On Some Aspects of Bayesian Survival Analysis

THESIS

FOR THE AWARD OF THE DEGREE OF

DOCTOR OF PHILOSOPHY

IN
STATISTICS

Submitted

By

Yasmin Khan

Under the supervision of

Professor Athar Ali Khan

Department of Statistics and Operations Research
Aligarh Muslim University
Aligarh- 202002
India
2016

List of Papers Published and Communicated

Published paper

1. Khan, Y., Akhtar, M. T., Sehla, R., and Khan, A. A. (2015). Bayesian modelling of forestry data with R using optimization and simulation tools. *Journal of Applied Analysis and Computation* **5**, 38-51.

2. Khan, Y., and Khan, A. A. (2014).Bayesian regression analysis of Weibull model with R and Bugs. *Statistical Methodologies and Applications* **2**, 131-140

3. Khan, Y., and Khan, A. A. (2013). Bayesian analysis of Weibull and log-normal survival models with censoring mechanism. *International Journal of Applied Mathematics* **26**, 671-683.

4. Khan, Y., and Khan, A. A. (2013) .Bayesian survival analysis of regression model using Weibull. *International Journal of Innovative Research and Science, Engineering and Technology* **2**, 7199-7204.

Paper communicated

1. Khan, Y., and Khan, A. A. (2015) . Generalized exponential model: A Bayesian study. *SpringerPlus,* (Springer Open).

2. Khan, Y., and Khan, A. A. (2016) . Bayesian Analysis of Lomax family with Laplace Approximation. *Applied Mathematics & Information Sciences Letters.*

Preface

The explosion of interest in Bayesian methods over the last ten to twenty years has been the result of the convergence of modern computing power and efficient Markov chain Monte Carlo (MCMC) algorithms for sampling from posterior distributions. The main aim of this thesis is to describe and illustrate the Bayesian modelling approach to the analysis of survival data. Emphasis is placed on the modeling of data and the interpretation of the results. Crucial to this is an understanding of the nature of the "incomplete" or "censored" data encountered. Understanding the censoring mechanism is important as it may influence model selection and interpretation. Yet, once understood and accounted for, censoring is often just another technical detail handled by the computer software, allowing emphasis to return to model building, assessment of model fit, assumptions and interpretation of the results.

The thesis consists of four chapters. Chapter 1 provides a quick and brief introduction of life time data. Some basic concepts and features of survival data are discussed. Following this, an introduction to Bayesian inference is also given in the same chapter emphasizing the need for the Bayesian approach to survival data analysis. It also consist of description of computer software which is to be used and illustrate the posterior summary of real survival data. In chapter 2, exponential distribution and its two extensions are discussed in Bayesian scenario. Analytic approximation and simulation tools are covered here, but most of the emphasis is on Markov chain based Monte Carlo methods including independent Metropolis algorithm and random-walk Matropolis algorithm, which are currently the most popular techniques. For analytic approximation, among various optimization algorithms three methods that are BFGS, Nelder-Mead and trust region methods are found to be the best. In this chapter analysis has been done by using two methods, i.e Nelder-Mead and trust region. Comparison of these two methods have also been made for the purpose of having better algorithm in hands. The main goal of this chapter is to compare the survival curves of two groups. The next chapter, Chapter 3 covers more complex problems, namely, exponentiated Weibull and its sub-models

have been used to model multiple regression problem for multiple myeloma pateints. Since, the data is having more than two regressor variables, the problem of variable selection is also handled through Bayesian analysis and caterpillar plots. Then the reduced form of regression model is analysed by using Nelder-Mead optimization method and independent Metropolis algorithm for the purpose of simulations. At the end of the chapter the exponentiated Weibull and its sub-models are compared by deviance and deviance information criteria (DIC). The last Chapter 4 is about the Bayesain analysis of Lomax distribution and recently introduced exponential Lomax and Weibull Lomax distributions. These distributions have been analysed analytically and using simulation tools. Again, model comparison has been made to pick the best model for the fitting of survival data.

The present thesis provides a comprehensive, short introduction to Bayesian theory used in the field of life time data. Only the most essential elements and some aspects of the specific topics in survival analysis are emphasized. Nevertheless, since this is intended as a comprehensive overview, detailed illustration using real survival data is also provided with emphasis on medical data analysis.

Acknowledgement

All praises and thanks to the Almighty Allah, the most beneficent and the most merciful, who in His infinite mercy has guided me to complete this PhD work.

Words and lexicons can not do full justice in expressing a deep sense of gratitude and indebtedness which flows from my heart towards my esteemed supervisor Prof. Athar Ali Khan, Department of Statistics and Operations Research, Faculty of Science, Aligarh Muslim University, Aligarh. Under his benevolence and able guidance, I learned and ventured to write. His inspiring benefaction bestowed upon me so generously and his unsparing help and precious advice have gone into the preparation of this thesis. He has been a beckon all long to enable me to present this work. I shall never be able to repay for all that I have gained from him.

I sincerely thank to Professor Qazi Mazhar Ali, Chairman, Department of Statistics and Operations Research A.M.U, Aligarh, for providing necessary facilities to carry out this thesis.

I also take this opportunity to thank every faculty members and non-teaching staff of the Department of Statistics and Operations Research, Faculty of Science, Aligarh Muslim University, Aligarh, who have contributed in anyway towards the completion of this task.

Many thanks are due to research scholars, Romana Shehla, Shagufta Yasmeen, Najrullah and Dr. Tanwir Akhtar for their help and encouragement in completion of my Ph.D. work. Also I would like thank my roommates Nishat shameem, Shagufta Yasmin for their moral support. I must not forget to mention Miss Azmi Naaz whose constant encouragement, like a sister helped me a lot in improving my work and for making my stay memorable and so enjoyable in the hostel.

I would also like to thank my colleagues Dr. Neha Gupta, Dr. Sanam Haseen and

Dr. Abdul Quddoos for their help and cooperation.

I would like to thank UGC for awarding me Maulana Azad National fellowship.

I must express my very profound gratitude to my mother Mrs. Aneesa, my father Mr. Kamruzzama Khan and my brothers, Asif Khan and Akib Khan for providing me with unfailing support and continuous encouragement throughout my years of study and through the process of research and writing this thesis. This accomplishment would not have been possible without them.

Finally, I owe a deep sense of gratitude to the authorities of Aligarh Muslim University, my Alma Mater, for providing me adequate research facilities.

Yasmin Khan

Table of Contents

List of Tables

Table **Page**

List of Figures

Figure **Page**

CHAPTER 1

Introduction

1.1 Introduction

Survival analysis and Bayesian methods are two active areas in the statistical literature. Ibrahim et al. (2001) have made an admirable accomplishment on the subject in well organised and easily accessible fashion. Survival analysis refers to a family of statistical method used to analyze duration of time until the occurrence of a well defined event, for example times to death of patients with certain disease, remission duration of certain disease in clinical trials, incubation times of certain disease, such as aids, hypertitis B, sars etc., failure times of certain manufactured products, life times of elderly in particular social programs, time taken by an individual to complete thesis, etc. It also refers to as "time to event analysis" which arises in a number of applied fields such as biology, engineering, medicine, demography, public health, economics and social science. Here our main focus is on the application of survival analysis to data arising from medical research, and for this reason much of the general discussion will be phrased in terms of the survival time of an individual patient from entry to a study until death. For many statisticians the statistical analysis of lifetime data has become a topic of considerable interest. There have been several text books written that address survival analysis from a freqentist perspective. This includes Lawless (1982), Cox and Oakes (1984), Fleming and Harrington (1991), Lee (1992), Anderson et al. (1993), Klein and Moeschberger (2003). The aim of this thesis is to describe and illustrate the Bayesian modeling approach to the analysis of survival data. It is well known that survival models are generally quite hard to fit, specially in the presence of complex censoring schemes.

With the use of various MCMC techniques, fitting complex survival models is fairly straightforward, and the availability of computer software packages ease the implementation greatly. In the frequentist paradigm, variance estimates, for example, usually require asymptotic arguments which can be quite complicated to derive and in some models are simply not available. Then there is always an issue of whether the sample size is large enough for the asymptotic approximation to be valid. In contrast, in the Bayesian framework, variance estimates, as well as any other posterior summary come out as a by-product of the simulation tools, and therefore are in principle, possible to obtain once samples from the posterior distribution are available. Historically, much of the survival analysis has been developed and applied in relation to cancer clinical trials in which the survival time is often measured from the date of randomization or commencement of therapy until death. The seminal papers by Peto et al. (1976, 1977) published in the British Journal of cancer describing the design, conduct and analysis of cancer trials provide a landmark in development and use of survival methods. On statistical science frontline, the development of statistical procedures and models for survival analysis exploded in the 1970s and 1980s were mostly derived from asymptotic methodology. By the late 1980s and early 1990s, survival analysis has established itself as the *defacto* standard method in biomedical research. The special and very common feature of survival data is that survival times are frequently censored. Accommodating and maximally utilizing the partial information from the censored observations, is the most challenging and also the most rewarding task in survival analysis as a unique field in mathematical statistics.

Survival data are generally described and modelled in terms of two related concepts, namely survival function and hazard function.

1.1.1 The survival function

The main quantity which employed to describe time-to-event phenomena is the survival function, the probability of an individual surviving beyond time t. Let T be a continuous nonnegative random variable representing the survival times of individuals in some population. All functions, unless stated otherwise, are defined over the interval $[0,\infty)$. Let $f(t)$ denote the probability density function of T and let the distribution function be

$$F(t) = P(T \leq t) = \int_0^t f(u)du$$

The probability of an individual's surviving till time t is given by the survivor function:

$$S(t) = 1 - F(t) = P(T > t) \tag{1.1}$$

It would be noted that $S(t)$ is a monotone decreasing function with $S(0) = 1$ and $S(\infty) = lim_{t\to\infty} S(t) = 0$.
The pth-quantile of the distribution of T is the value t_p such that

$$F(t_p) = P(T \leq t_p) = p.$$

That is, $t_p = F^{-1}(p)$. The pth-quantile is also referred to as the $100 \times pth$ percentile of the distribution.

1.1.2 Basic properties of survival function

In this thesis, exponential and Weibull distribution will be discussed as in the special case of generalized exponential and exponentiated Weibull distribution, respectively. So, in this section survival curves of Weibull distribution could be seen in Figure 1.1. Many types of survival curves can be shown but the important

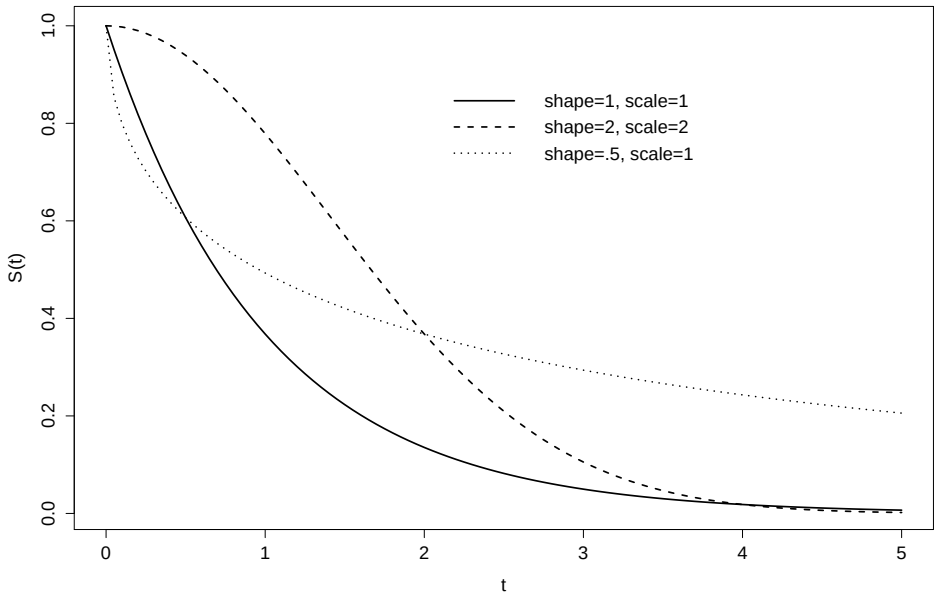

Figure 1.1: *Weibull survival functions for different values of shape and scale.*

point to note is that they all have the same basic properties. They are monotone, nonincreasing functions equal to one at zero and zero as the time approaches infinity. Thus,

1. $S(0) = 1$, $S(\infty) = 0$;

2. $S(x)$ is non-increasing.

Their rate of decline, of course, varies according to the risk of experiencing the event at time t but it is difficult to determine the essence of a failure pattern by simply looking at the survival curve. Nevertheless, this quantity continues to be a popular description of survival in the applied literature and can be very useful in comparing two or more mortality patterns. Next, we present the second most important quantity for the analysis of survival data, i.e, hazard function.

1.1.3 The hazard function

The hazard function gives the instantaneous failure rate at t given that the individual has survived up to time t. Hazard function is particularly useful in determining the appropriate failure mechanism. The hazard rate is defined as

$$h(t) = \lim_{\Delta t \to 0} \frac{P[t \leq T < t + \Delta t | T \geq t]}{\Delta t} \tag{1.2}$$

If T is a continuous random variable,

$$h(t) = \frac{f(t)}{S(t)} \tag{1.3}$$

In particular, $h(t)\Delta t$ is the approximate probability of failure in $[t, t + \Delta t)$, given survival upto time t. The hazard function is also referred to as the risk or mortality rate. The functions $f(t)$, $F(t)$, $S(t)$, and $h(t)$ give mathematically equivalent specifications of the distribution of T. It is easy to derive expressions for $S(t)$ and $f(t)$ in terms of $h(t)$. Since $f(t) = -\frac{d}{dt}S(t)$, Equation 1.3 implies that

$$h(t) = -\frac{d}{dt} log(S(t)) \tag{1.4}$$

Now integrating both sides of Equation 1.4, then exponentiating,

$$S(t) = exp\left(-\int_0^t h(u)du\right) \tag{1.5}$$

Some generic types of hazard rates are plotted in Figure 1.2. For example, one may believe that the hazard rate for the occurrence of a particular event is increasing, decreasing, constant or possessing some other characteristic which describes the failure mechanism. Models with increasing hazard rates may arise when there is natural aging or wear. Decreasing hazard functions are much less common but find occasional use when there is a very early likelihood of failure, such as in certain types of electronic devices or in patients experiencing certain types of transplants.

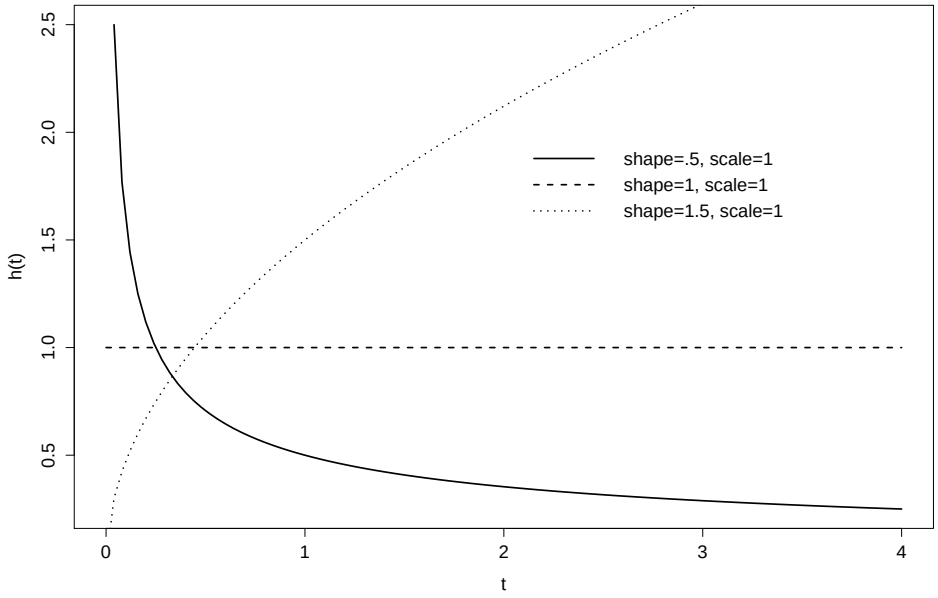

Figure 1.2: *Weibull hazard functions for different values of shape and scale.*

1.2 Features of survival data

There are mainly two important features of survival data

1. Censoring mechanism
2. Skewness

1.2.1 Skewness

An important feature of survival data is its skewness. Consequently, normal theory of linear models does not work and models like Weibull, log normal and log-logistic are commonly used.

1.2.2 Censoring mechanism

Censoring occurs when we have some information about individual survival time, but we do not know the survival time exactly. General censoring process has been studied by several authors, including Cox (1975), Efron (1977), Kalbfleisch and MacKay (1978), Kalbfleisch and Prentice (1980, 2002). An actual survival time can also be regarded as censored when death is from a cause that is known to be unrelated to the treatment. However, it can be difficult to be sure that the death is not related to a particular treatment that the patient is receiving. For

example, consider a patient in a clinical trial to compare alternative therapies for prostatic cancer who experiences a fatal road traffic accident. The accident could have resulted from an attack of dizziness, which might be a side effect of the treatment to which that patient has been assigned. If so, the death is not unrelated to the treatments. In circumstances such as these, the survival time until death from all causes, or the time to death from causes other than the primary condition for which the patient is being treated, might also be subjected to a survival analysis. In each of these situations, a patient who entered a study at time t_0 and die at time $t_0 + t$. However, t is unknown, either, because the individual is still alive or because he or she has been lost to follow up. If the individual was last known to be alive at time $t_0 + c$, the time c is called a censored survival time. More clearly the pattern of censoring is defined in the following section.

1.2.3 Pattern of censoring

1. **Right censoring:** The most frequent type of censoring encountered in survival analysis data is right censored data. This refers to individuals who are followed from the beginning of the study until a time point where they are lost during the follow-up that is, we know the exact time of entry of a patient but do not have the availability of exact time of death. For example, remission duration from a clinical trial for acute leukemia.

2. **Left censoring:** This censoring refers to cases where the exact time when the patient entered the study is unknown but the exact time of death is available. For example, childhood learning. time-to-event: the age at which a child learns to accomplish certain tasks in children learning centers. Left censoring occurs if children can already perform the tasks when they start their study at the centers.

3. **Interval censoring:** This type of censoring refers to cases where both the exact times of death and the entry into the study of a patient are unknown. This type of censored data inform us that the individual was alive at specific time points, so we know that the survival time was greater than a specific value t. For example, Time to cosmetic deterioration of breast cancer patients.

1.2.4 The likelihood function for censored data

Suppose that there are n subjects under study, and that associated with the ith individual is a survival time t_i and a censoring time t_{c_i}. The t's are assumed to be independent and identically distributed with density $f(t)$ and survival function

$S(t)$. The exact survival time t_i of an individual will be observed only if $t_i \leq t_{c_i}$. The data in this framework can be represented by the n pairs of random variables (Y_i, δ_i), where

$$Y_i = min(t_i, t_{c_i})$$

It is useful to introduce a binary random variable δ which indicates if a failure time is observed or censored,

$$\delta_i = \begin{cases} 1 & \text{if } t_i \leq t_{c_i}, \\ 0 & \text{if } t_i > t_{c_i}. \end{cases} \tag{1.6}$$

Now, for the construction of likelihood function for the censored data we need to calculate the joint likelihood of the pair (Y_i, δ_i). By likelihood we mean the rubric which regards the density as a function of the parameter for a given (fixed) value (y_i, δ_i). For $y < t_c$, $P(Y \leq y) = P(T \leq y) = F(y)$ and $P(\delta = 1 | Y \leq y) = 1$. Therefore, the likelihood for $Y = y < t_c$ and $\delta = 1$ is the density $f(y)$. For $y = t_c$ and $\delta = 0$, the likelihood for this event is the probability $P(\delta = 0, Y = t_c) = P(T > t_c) = S(t_c)$.

We can combine these two expressions into one single expression $f(y)^{\delta} S(t_c)^{1-\delta}$. As usual, the likelihood function of a random sample is defined as the product of the densities of the individual observations. That is, the likelihood function for the n iid random pairs (Y_i, δ_i) is given by

$$L = \prod_{i=1}^{n} f(y_i)^{\delta_i} S(t_c)^{1-\delta_i}. \tag{1.7}$$

1.3 Models for survival data

There are two main approaches for regression modeling of lifetime data. One uses time transformations, assuming that the effect of covariates is equivalent to altering the rate at which time passes, called log-location scale regression model or accelerated failure time models. The second approach adopts specifications of the way that the covariates affect the hazard function for T, called proportional hazard models. Out of these two models only proportional hazard regression model will be discussed in the thesis and its general theory will be discussed in the next section.

1.3.1 Proportional hazard models

A proportional hazard family is a class of models with the property that different individuals have hazard functions which are proportional to one another. That is, the ratio $h(t|x_1)/h(t|x_2)$ of the hazard functions for two individuals, with regression vector x_1 and x_2, does not vary with t. This implies that hazard function of T,

given x, can be written in the form

$$h(t|x) = h_0(t)g(x) \tag{1.8}$$

Both h_0 and g may involve unknown parameters; $h_0(t)$ is the baseline hazard function. A particularly useful family of models is obtained from a univariate lifetime model with hazard function $h_0(t)$ by defining

$$h(t|x) = h_0(t)exp(x^T\beta) \tag{1.9}$$

where $x\beta = x_1\beta_1 + \ldots + x_p\beta_p$ and the $\beta_i's$ are unknown regression coefficients. This model is natural and sufficiently flexible for many purposes. Since, $\exp(x\beta)$ is always positive, $h(t|x)$ is automatically nonnegative for all x and β.

Now, the effect of x on the survivor function in the family of models of Equation 1.8. Since

$$S(t|x) = exp\left(-\int_0^t h(u|x)dx\right)$$

it follows that the survivor function of T, given x, is

$$S(t|x) = S_0(t)^{g(x)} \tag{1.10}$$

where

$$S_0(t) = exp\left(-\int_0^t h_0(u)du\right)$$

is the baseline survivor function for an individual with $g(x) = 1$

The proportional hazard model used in the thesis are discussed as

1. Generalized exponential regression model.
2. Exponential extension regression model.
3. Exponential regression model.

The proportional hazard regression model is

Model I: $h(t|x) = h_0(t)exp(x^T\beta)$

Model II: $h(t|x) = \dfrac{\alpha/\lambda(1-exp(-t/\lambda))^{\alpha-1}exp(-t/\lambda)}{1-(1-exp(-t/\lambda))^{\alpha}}exp(x^T\beta)$

Model III: $h(t|x) = \alpha/\lambda(1+t/\lambda)^{\alpha-1}exp(x^T\beta)$

Model I represent the general form of proportional regression hazard model, Model II is the generalized exponential, Model III is the exponential extension proportion hazard model.

1.4 The Bayesian paradigm

Bayesian statistical analysis is based on the premise that all uncertainty should be modeled using probabilities and that statistical inferences should be logical conclusions based on the laws of probability.

The field of statistics has long embraced the concept of probability models for data. Such models typically involve parameters that are presumed to be related to characteristics of the sampled populations. These parameters can range from few in number with simple interpretations to an uncountable number. Parameters can never be known with absolute certainty unless we sample the entire population. Moreover, parameters may not have physical interpretations since, inevitably, models rarely are precisely true.

Given a statistical model for the data, the Bayesian approach mandates an additional probability model for all unknown parameters in the data model. Our approach is to model this uncertainty about the parameters using scientific expert information. This information is called "prior" information, or information that has been collected a priori. Expert information must be obtained independently of the data being analyzed. One way to guarantee that scientific input about model parameters is independent of the data is to acquire that information before the data have been collected. However, despite the a priori terminology, such information is often not literally obtained prior to the collection of data. Our experience is that it is generally possible to obtain independent information from sources such as existing literature or colleagues of the scientists who collected the current data. Throughout the book, we use the word "prior" partly for simplicity of exposition and partly for historical reasons, but it is understood that prior information is simply information obtained independently of the current data.

There is a vast literature on Bayesian statistics. Four foundational works are De Finetti (1974, 1975), Jeffreys (1961), and Savage (1954). Good elementary introductions to the subject are Lindley (1971) and Berry (1996). Early efforts to make Bayesian methods accessible for data analysis were made by Raiffa and Schlaifer (1961), Zellner (1971), and Box and Tiao (1973). The important topic of

Bayesian prediction was presented in Aitchison and Dunsmore (1975) and Geisser (1993). Bayesian decision theory and more theoretical aspects of Bayesian inference were presented in DeGroot (1970) and Berger (1985). Modern Bayesian data analysis methods based on Markov chain Monte Carlo methods are presented in Gelman et al. (1995, 2004), Carlin and Louis (1996, 2008), Congdon (2001, 2003) and Marin and Robert (2007). Recent theoretical treatments are found in Roberts (2007) and Bernardo and Smith (2000). The most popular interpretations and approaches are objective Bayesian inference Berger (2006) and subjective Bayesian inference Anscombe and Aumann (1963), Goldstein (2006). Objective Bayesian inference is often associated with Bayes and Price (1763), Laplace (1814), and Jeffreys (1961). Subjective Bayesian inference is often associated with Ramsey (1926), De Finetti (1931), and Savage (1954). The first major event to bring about the rebirth of Bayesian inference was De Finetti (1937).

Although Bayesian methodology allows every data analyst their own prior distribution, we believe that it remains consistent with the practice of science. For large amounts of data, scientists with different prior beliefs should ultimately agree after (separately) combining the data with their prior information. At least, this should happen for anyone with a "reasonable" prior. On the other hand, insufficient data can result in (continued) discrepancies of opinion about relevant scientific questions. In the real world, that is how science works. More philosophically, Bayesian statistics appears to be the only logically consistent method of making statistical inferences, although not the only useful one, (Christensen et al. 2011).

1.4.1 Statistical model

Statistical models typically involve multiple observations (random variables), say, $y_1, \ldots, y_n$. Dealing with these is facilitated by writing them collectively as a vector of observations, say $y = (y_1, \ldots, y_n)'$ where the $'$ indicates transposing of the row vector so that y is an $n \times 1$ matrix. Typically, the observations are collected independently given the parameters of the model. Denote the parameters $\theta = (\theta_1, \ldots, \theta_r)'$. In many simple problems $r = 1$.

Bayesian statistics typically begins with prior information about the state of nature θ that is embodied in the prior density $p(\theta)$. It then uses Bayes' Theorem and the random data y, with sampling density $p(y|\theta)$, to update this information into a posterior density $p(\theta|y)$ that incorporates both the prior information and the data. Specifically, Bayes' Theorem tells us that

$$p(\theta|y) = \frac{p(y|\theta)p(\theta)}{\int p(y|\theta)p(\theta)d\theta}$$

1.4.2 Three important components of Bayesian inference

The three important components of Bayesian Inference are

Prior distribution $p(\theta)$ is the set of prior distributions for parameter set θ, and uses probability as a means of quantifying uncertainty about θ before taking the data into account.

Likelihood $p(y|\theta)$ is the likelihood or likelihood function, in which all variables are related in a full probability model.

Posterior distribution $p(\theta|y)$ is the joint posterior distribution that expresses uncertainty about parameter set θ after taking both the prior and the data into account. If parameter set θ is partitioned into a single parameter of interest ϕ and the remaining parameters are considered nuisance parameters, then $p(\phi|y)$ is the marginal posterior distribution.

1.4.3 Prior distribution

Implementation of the Bayesian approach depends on a willingness to assign probability distributions not only to data variables like y, but also to parameters like θ. Such a requirement may or may not be consistent with the usual long-run frequency notion of probability. For example, if

$$\theta = \text{true probability of success for a new surgical procedure,}$$

then it is possible (at least conceptually) to think of θ as the limiting value of the observed success rate as the procedure is independently repeated again and again. But if

$$\theta = \text{true proportion of U.S. men who are HIV-positive,}$$

the long-term frequency notion does not apply; it is not possible to even imagine "running the HIV epidemic over again" and reobserving θ. Moreover, the randomness in θ does not arise from any real-world mechanism; if an accurate census of all men and their HIV status were available, θ could be computed exactly. Rather, here θ is random only because it is unknown to us, though we may have some feelings about it (say, that $\theta = .05$ is more likely than $\theta = .50$). Bayesian analysis is predicated on such a belief in subjective probability, wherein we quantify whatever feelings (however vague) we may have about θ before we look at the data y in a distribution $p(\theta)$. This distribution is then updated by the data via Bayes Theorem.

Historically, a major impediment to widespread use of the Bayesian paradigm has been that determination of the appropriate form of the prior $p(\theta)$ (and perhaps the hyperprior h) is often an arduous task. Typically, these distributions are specified based on information accumulated from past studies or from the opinions of subject-area experts. In order to streamline the elicitation process, as well as simplify the subsequent computational burden, experimenters often limit this choice somewhat by restricting $p(\theta)$ to some familiar distributional family. An even simpler alternative, available in some cases, is to endow the prior distribution with little informative content, so that the data from the current study will be the dominant force in determining the posterior distribution, (Carlin and Louis, 2008).

1.4.4 Noninformative priors

As alluded to earlier, often no reliable prior information concerning θ exists, or an inference based solely on the data is desired. At first, it might appear that Bayesian inference would be inappropriate in such settings, but this conclusion is a bit hasty. Suppose we could find a distribution $p(\theta)$ that contained "no information" about θ in the sense that it did not favor one θ value over another (provided both values were logically possible). We might refer to such a distribution as a noninformative prior for θ, and argue that all of the information resulting in the posterior $p(\theta|y)$ arose from the data, and hence all resulting inferences were completely objective, rather than subjective. Such an approach is likely to be important if Bayesian methods are to compete successfully in practice with their popular likelihood-based counterparts (e.g., maximum likelihood estimation). But is such a "noninformative approach" possible?

In some cases, the answer is an unambiguous "yes." For example, suppose the parameter space is discrete and finite, i.e., $\Theta = \theta_1, \ldots, \theta_n$. Then clearly the distribution

$$p(\theta_i = 1/n),\ i = 1, 2, \ldots, n$$

does not favor any one of the candidate θ values over any other and, as such, is noninformative for θ. If instead we have a bounded continuous parameter space, say $\Theta = [a, b]$, $-\infty < a < b < \infty$, then the uniform distribution

$$p(\theta) = 1/(b-a), a < \theta < b,$$

is arguably noninformative for θ.
When we move to unbounded parameter spaces, the situation is even less clear. Suppose that $\Theta = (-\infty, \infty)$. Then the appropriate uniform prior would appear to be

$$p(\theta) = c, \text{ any } c > 0$$

But this distribution is improper, in that $\int p(\theta)d\theta = \infty$, and hence does not appear appropriate for use as a prior. But even here, Bayesian inference is still possible if the integral with respect to θ of the likelihood $p(y|\theta)$ equals some finite value K, (Carlin and Louis, 2008).

1.4.5 Other prior construction methods

There are a multitude of other methods for constructing priors, both informative and noninformative; see Berger (1985, Chapter 3) for an overview. The only one we shall mention is that of using the marginal distribution $p(y) = \int p(y|\theta)p(\theta)d\theta$. Here, we choose the prior $p(\theta)$ based on its ability to preserve consistency with the marginal distribution of the observed data. Unlike the previous approaches in this section, this approach uses not only the form of the likelihood, but also the actual observed data values to help determine the prior. We might refer to this method as empirical estimation of the prior, and this is, in fact, the method that is used in empirical Bayes (EB) analysis.

Strictly speaking, empirical estimation of the prior is a violation of Bayesian philosophy: the subsequent prior-to-posterior updating would "use the data twice" (first in the prior, and again in the likelihood). The resulting inferences from this posterior would thus be "overconfident." Indeed, EB methods that ignore this fact are often referred to as naive EB methods, (Carlin and Louis, 2008).

1.4.6 Likelihood

In order to complete the definition of a Bayesian model, both the prior distributions and the likelihood must be approximated or fully specified. The likelihood, likelihood function, or $p(y|\theta)$, contains the available information provided by the sample. The likelihood is

$$p(y|\theta) = \prod_{i=1}^{n} p(y_i|\theta)$$

The data y affects the posterior distribution $p(\theta|y)$ only through the likelihood function $p(y|\theta)$. In this way, Bayesian inference obeys the likelihood principle, which states that for a given sample of data, any two probability models $p(y|\theta)$ that have the same likelihood function yield the same inference for θ.

1.4.7 Likelihood function of a parameterized model

In non-technical parlance, "likelihood" is usually a synonym for "probability", but in statistical usage there is a clear distinction: whereas "probability"allows us to predict unknown outcomes based on known parameters, "likelihood" allows us to

estimate unknown parameters based on known outcomes. In a sense, likelihood can be thought a reversed version of conditional probability. Reasoning forward from a given parameter θ, the conditional probability of y is the density $p(y|\theta)$. With θ as a parameter, here are relationships in expressions of the likelihood function

$$L(\theta|y) = p(y|\theta) = f(y|\theta)$$

For example, in a Bayesian linear regression with an intercept and two independent variables, the model may be specified as

$$y_i \sim N(\mu_i, \sigma^2)$$

$$\mu_i = \beta_1 + \beta_2 X_{i,1} + \beta_3 X_{i,2}$$

The dependent variable y, indexed by $i = 1, \ldots, n$, is stochastic, and normally-distributed according to the vector μ, and variance σ^2. Vector μ is an additive, linear function of a vector of regression parameters, β, and the design matrix X. Since y is normally-distributed, the probability density function (PDF) of a normal distribution will be used, and is usually denoted as

$$p(y) = \frac{1}{\sqrt{2\pi}\,\sigma} \exp\left[(-\frac{1}{2\sigma^2})(y_i - \mu_i)^2\right];\ \ y\,\varepsilon\,(-\infty, \infty)$$

By considering a conditional distribution, the record-level likelihood in Bayesian notation is

$$p(y_i|\Theta) = \frac{1}{\sqrt{2\pi}\,\sigma} \exp\left[(-\frac{1}{2\sigma^2})(y_i - \mu_i)^2\right];\ \ y\,\varepsilon\,(-\infty, \infty)$$

In both theory and practice, and in both frequentist and Bayesian inference, the log-likelihood is used instead of the likelihood, on both the record and model-level. The model-level product of record-level likelihoods can exceed the range of a number that can be stored by a computer, which is usually affected by sample size. By estimating a record-level log-likelihood, rather than likelihood, the model-level log-likelihood is the sum of the record-level log-likelihoods, rather than a product of the record-level likelihoods.

$$\log[p(y|\theta)] = \sum_{i=1}^{n} \log[p(y_i|\theta)]$$

rather than

$$p(y|\theta) = \prod_{i=1}^{n} p(y_i|\theta)$$

1.4.8 Posterior distribution

In addition to using the sampling density $p(y|\theta)$, Bayesians incorporate prior knowledge about θ through a density $p(\theta)$. The joint density of θ and y is then

$$p(\theta, y) = p(y|\theta)p(\theta)$$

Integrating out θ, the marginal density of y is

$$p(y) = \int p(y|\theta)p(\theta)d\theta$$

The marginal distribution of the data is sometimes called the marginal predictive distribution. By definition, the conditional density of θ given y is

$$p(\theta|y) = \frac{p(y,\theta)}{p(y)}$$

Bayes' Theorem tells us that

$$p(\theta|y) = \frac{p(y|\theta)p(\theta)}{\int p(y|\theta)p(\theta)d\theta}$$

The integral in the denominator is r dimensional. The posterior density $p(\theta|y)$ is a function of θ , so the denominator $f(y)$ is merely a constant. In other words, from probability theory,

$$1 = \int p(\theta|y)d\theta,$$

so the denominator is whatever constant is needed to make $p(y|\theta)p(\theta)$ integrate to 1. We often write

$$p(\theta|y) \propto p(y|\theta)p(\theta),$$

To a Bayesian, the best information one can ever have about θ is to know the posterior density $p(\theta|y)$. Nonetheless, it is often convenient to summarize the posterior information. Most summaries involve integration, which we will typically perform by computer simulation. In multidimensional problems, the marginal posterior density of, say, θ_1 and θ_2 is

$$p(\theta_1,\theta_2|y) = \int \dots \int p(\theta|y)d\theta_3 \dots d\theta_r.$$

1.4.9 Bayesian inference versus frequentist inference

There is a growing interest in the use of Bayesian methods in various statistical and other fields also. In this section we will begin with a basic and brief introduction of Bayesian inference and its advantage over frequentist inference. Throughout the thesis, the broad uses of Bayesian methods for a variety of inferential and statistical task has been explored.

In statistical inference, there are two broad categories of interpretation of probability, which lead to two broad categories of inference:

- Bayesian inference
- Frequentist inference

Bayesian methods have certain advantages over its counterpart. Some of them are reported as under:

- Bayesian inference can avoid problems with model identification by manipulating prior distributions (usually in complex models). Frequentist inference with any numerical approximation algorithm does not have prior distributions, and can become stuck in regions of at density, causing problems with model identification.
- Bayesian inference considers the data to be fixed (which it is), and parameters to be random because they are unknowns. Frequentist inference considers the unknown parameters to be fixed, and the data to be random, estimating not based on the data at hand, but the data at hand plus hypothetical repeated sampling in the future with similar data. "The Bayesian approach delivers the answer to the right question in the sense that Bayesian inference provides answers conditional on the observed data and not based on the distribution of estimators or test statistics over imaginary samples not observed" (Rossi et al. 2005, p. 4).
- Bayesian inference allows informative priors so that prior knowledge or results of a previous model can be used to inform the current model.
- Bayesian inference estimates p(hypothesis|data). In contrast, frequentist inference estimates p(data|hypothesis). Even the term 'hypothesis testing' suggests it should be the hypothesis that is tested, given the data, not the other way around.
- Bayesian inference includes uncertainty in the probability model, yielding more realistic predictions. Frequentist inference does not include uncertainty of the parameter estimates, yielding less realistic predictions.
- Bayesian inference uses prior distributions, so more information is used and 95% probability intervals of posterior distributions should be narrower than 95% confidence intervals of frequentist point-estimates.
- Bayesian inference uses probability intervals (quantile-based, highest posterior density, or preferably lowest posterior loss) to state the probability that θ is between two points. Frequentist inference uses confidence intervals, which must be interpreted with probability of zero or one that θ is in the region, and the frequentist never knows whether it is or is not, but can only say that if 100 repeated samples were drawn in the future, that it would be in the region for 95 samples.

- Bayesian inference via MCMC or PMC is unbiased with respect to sample size and can accommodate any sample size no matter how small. Frequentist inference becomes more biased as sample size decreases from infinity, and is often wildly biased with small samples, so minimum sample size is an issue. Conversely, frequentist inference with large sample sizes biases p-values to indicate that insignificant effects are significant.

- Bayesian inference via MCMC or PMC uses exact estimation with respect to sample size. Frequentist inference uses approximate estimation that relies on asymptotic theory.

- Bayesian inference with proper priors is immune to singularities and near-singularities with matrix inversions, unlike frequentist inference.

1.5 Bayesian computation in survival analysis

As we discussed in the above section that skewness of survival data is modeled by the distribution like Weibull, lognormal and loglogistic instead of normal. However, in Bayesian scenario analysis of such models are not simple in frequentist framework. Several adhock methods are used in the analysis. Bayesian approach assimilates these methods in a single framework, called Bayes' rule or Bayes Theorem. Bayesian methods are very well suited for survival data and provides a parsimonious description of survival data analysis and a computational framework for model estimation, selection and model comparison. Bayesian survival analysis consists of likelihood and prior information and It generates conclusions in the form of posterior distribution. Since Bayesian methods are becoming quite common and popular, we will implement this approach on some aspects of survival data analysis. Ibrahim et al. (2001) identified two key advantages. First, survival models are generally very difficult to fit, due to the complex likelihood functions to accommodate censoring. A Bayesian approach may help using `MCMC` techniques and there is available software e.g. `LaplacesDemon` and `BUGS`. Second, The Bayesian paradigm can incorporate prior information in a natural way by using historical information, e.g. from clinical trials. This thesis consist of analysis of survival data with and without censoring mechanism in a complete Bayesian environment. The previous section will describes some of the advantages of Bayesian approach over frequentist. In situations where the posterior distribution is not a standard functional forms, there are numbers of methods available, i.e Laplace approximation, independent Metropolis algorithm which are used for computing integrals and simulating from a general posterior distribution. This thesis focused on the use of

computational methods that are applicable to high-dimensional Bayesian problems that arise in survival analysis.

1.5.1 Numerical approximation

The technical problem of evaluating quantities required for Bayesian inference typically reduces to the calculation of a ratio of two integrals (Bernardo and Smith 2000, p. 339). In all cases, the technical key to the implementation of the formal solution given by Bayes' theorem is the ability to perform a number of integrations (Bernardo and Smith 2000, p. 340). Except in certain rather stylized problems, the required integrations will not be feasible analytically and, thus, efficient approximation strategies are required.

There are too many different types of numerical approximation algorithms in Bayesian inference to cover in any detail in this thesis. However, a few important methods are covered. Since, Laplace approximation is a very powerful tool of analytic approximation, therefore, it has been discussed and implemented throughout the thesis. Next important tool of Bayesian analysis is the simulation. Among the MCMC methods, a few important ones are random walk Metropolis and independent Metropolis algorithm which are discussed and implemented throughout thesis. The `R` package, `LaplacesDemon` deals with both analytic and simulation tools. The function `LaplaceApproximation` deals with analytic approximation whereas `LaplacesDemon` function implements simulation algorithms. Now, we summarize some methods for computing integrals.

1.5.2 Normal approximation

If the posterior distribution $p(\theta|y)$ is unimodal and roughly symmetric, it can be convenient to approximate it by a normal distribution; that is, the logarithm of the posterior density is approximated by a quadratic function of θ.

Here we consider a quadratic approximation to the log-posterior density that is centered at the posterior mode (which in general is easy to compute using off-the-shelf optimization routines).

A Taylor series expansion of $\log p(\theta|y)$ centered at the posterior mode, $\hat{\theta}$ (where θ can be a vector and $\hat{\theta}$ is assumed to be in the interior of the parameter space), gives

$$\log p(\theta|y) = log\ p(\hat{\theta}|y) + \frac{1}{2}(\theta-\hat{\theta})^T \left[\frac{d^2}{d\theta^2}\log p(\theta|y)\right]_{\theta=\hat{\theta}} + (\theta-\hat{\theta}) + \ldots, \tag{1.11}$$

where the linear term in the expansion is zero because the log-posterior density has zero derivative at its mode. Considering Equation 1.11 as a function of θ, the first term is a constant, whereas the second term is proportional to the logarithm of a normal density, yielding the approximation,

$$p(\theta|y) \approx N(\hat{\theta}, [I(\hat{\theta})]^{-1}), \tag{1.12}$$

where $I(\theta)$ is the observed information,

$$I(\theta) = -\frac{d^2}{d\theta^2} \log p(\theta|y) \tag{1.13}$$

If the mode, $\hat{\theta}$, is in the interior of parameter space, then the matrix $I(\hat{\theta})$ is positive definite, Gelman et al. (1995, 2004).

1.5.3 The Laplace approximation

Laplaceapproximation technique was originally presented in Laplace 1774 (reprinted Stigler 1986); the most frequently cited paper on the subject is the rather more recent one by Tierney and Kadane (1986). Laplace Approximation dates back to Laplace (1774, 1814), and is used to approximate the posterior moments of integrals. Extensions and refinements were made by Kass (1989) and Wong and Li (1992). Geweke (1989) discusses modal approximations for importance sampling and proposes the k-variate split normal density as an improved approximation for asymmetric posterior densities. Specifically, the posterior mode is estimated for each parameter, assumed to be unimodal and Gaussian. As a Gaussian distribution, the posterior mean is the same as the posterior mode, and the variance is estimated. Laplace Approximation is a family of deterministic algorithms that usually converge faster than MCMC, and just a little slower than Maximum Likelihood Estimation (MLE) (Azevedo-Filho and Shachter 1994). Laplace Approximation shares many limitations of MLE, including asymptotic estimation with respect to sample size.

Many posterior summaries are expressible in terms of integrals. For example, suppose one is interested in posterior mean of a function $g(\theta)$. This mean is expressible as a ratio of integrals,

$$E(g(\theta)|y) = \frac{\int g(\theta)p(\theta)p(y|\theta)d\theta}{\int p(\theta)p(y|\theta)d\theta} \tag{1.14}$$

If we are interested in posterior probability that $g(\theta)$ falls in a set A, we wish to compute,

$$P(g(\theta)\epsilon A|y) = \frac{\int_{g(\theta)\epsilon A} p(\theta)p(y|\theta)d\theta}{\int p(\theta)p(y|\theta)d\theta} \tag{1.15}$$

Let us assume that $h(\theta) = log p(\theta)p(y|\theta)$, then integrals involved can be approximated by making use of Taylor's series expansion around posterior mode, $\hat{\theta}$ and integrands involving quadratic terms can be approximated by a multivariate normal distribution. This gives laplace approximation

$$h(\theta) \approx h(\hat{\theta}) + (\theta - \hat{\theta})' h''(\hat{\theta})(\theta - \hat{\theta})/2,$$

where $h''(\hat{\theta})$ is the Hessian of the log density evaluated at the mode. Using this expansion, the posterior density is approximated by a multivariate normal density with mean $\hat{\theta}$ and variance-covariance matrix

$$V = (-h''\hat{\theta})^{-1}$$

In addition, this approximation allows one to analytically integrate out θ from the joint density and obtain the following approximation to the prior predictive density,

$$p(y) = \approx (2\pi)^{d/2} p(\hat{\theta})p(y|\hat{\theta})|-h''(\hat{\theta})|^{1/2},$$

where d is the dimension of θ. To apply this approximation, one needs to find the mode of the posterior density of θ. One general purpose of optimization algorithm for finding this mode is provided by Newton's method. Suppose one has guess at the posterior mode θ^0. If θ^{t-1} is the estimate at the mode at the $t-1$ iteration of the algorithm, then the next iteration is given by,

$$\theta^t = \theta^{t-1} - [h''(\theta^{t-1})]^{-1} h'(\theta^{t-1}),$$

where $h'(\theta^{t-1})$ and $h''(\theta^{t-1})$ are the gradient and Hessian of the log density evaluated at the current guess at the mode. One continues these iterations until convergence. There are many alternative algorithms available for finding the posterior mode. In this thesis Trust region algorithm of Nocedal and Wright (1999) and Nelder and Mead (1965) method will be used. These two methods will be discussed in Section 1.8.2 also implementation has been done in the second chapter and a comparison has also been made. Since Nelder-Mead is a derivative free method and less sensitive for guess values, it works well in most of the practical situations. So, this method has been used throughout the thesis as an argument for optimization method in `LaplaceApproximation` function. However, it has been found that TR method works better than NM.

1.5.4 Monte Carlo method for computing integrals

A general approach for summarizing a posterior distribution is based on simulation. Suppose that θ has a posterior density $p(\theta|y)$ and one interested in learning about

a particular function of the parameters $h(\theta)$. The posterior mean of $h(\theta)$ is given by,

$$E(h(\theta)|y) = \int h(\theta)p(\theta|y)d\theta.$$

Suppose we are able to simulate an independent sample $\theta_1,\ldots,\theta_m$ from the posterior density. Then the Monte Carlo estimate at the posterior mean is given by the sample mean

$$\bar{h} = \frac{\sum_{j=1}^{m} h(\theta^j)}{m}$$

The associated simulation standard error of this estimate is estimated by

$$se_{\bar{h}} = \sqrt{\frac{\sum_{j=1}^{m}(h(\theta^j) - \bar{h})^2}{(m-1)m}}$$

The use of Monte Carlo methods is wide-spread in statistics and science in general. Rubinstein and Kroese (2008) cover Monte Carlo methods for a wide variety of statistical problems, Robert and Casella (2004) includes more coverage of Bayesian applications and MCMC methods as well.

1.5.5 Sampling importance resampling

A general purpose algorithm for simulating random draws from a given probability distribution is sampling importance resampling (SIR). SIR is a method of obtaining a simulated sample from the posterior density. It was introduced in Gordon et. al (1993), and is the original particle filtering algorithm. A distribution is approximated with importance weights, which are approximations to the relative posterior densities of the particles, and the sum of the weights is one. In this terminology, each sample in the distribution is a 'particle'. In SIR, the expectation of a function can be approximated as a weighted average. SIR is a sequential or recursive form of importance sampling. As in importance sampling, the expectation of a function can be approximated as a weighted average. The optimal proposal distribution is the target distribution.

As before, we simulate m draws from the proposal density q denoted by $\theta^1,\ldots,\theta^m$ and compute the weights $w(\theta^j) = \frac{p(\theta^j|y)}{q(\theta^j)}$. Convert the weights to the probabilities by using the formula

$$q^j = \frac{w(\theta^j)}{\sum_{j=1}^{m} w(\theta^j)}$$

Suppose we take a new sample $\theta^{*1},\ldots,\theta^{*m}$ from the discrete distribution over $\theta^1,\ldots,\theta^m$ with respective probabilities $q^1,\ldots,q^m$. Then the θ^{*j} will be approximately distributed according to the posterior distribution p. This method, called

sampling importance resampling or SIR for short, is a weighted bootstrap procedure where we sample with replacement from the sample θ^j with unequal sampling probabilities, Albert (2009).

1.6 Markov Chain Monte Carlo

Markov chain Monte Carlo (MCMC) techniques were rediscovered in early **1990**. MCMC methods are not new, as they were introduced into physics in 1953 in a simplified version by Metropolis and his associates (Metropolis et al., 1953). Intermediate landmark publications include the generalization of Metropolis algorithm by Hastings (1970) and development of the Gibbs sampler by Geman and Geman (1984). Nevertheless, it took about 35 years until MCMC methods were rediscovered by Bayesian scientists (Tanner and Wong, 1987; Gelfand et al., 1990; Gelfand and Smith, 1990) and became one of the main computational tools in modern statistical inference.

Markov chain Monte Carlo techniques enabled quantitative researchers to use highly complicated models and estimate the corresponding posterior distributions with accuracy. In this way, MCMC methods have greatly contributed to the development and propagation of Bayesian theory. Extensive details of the use of MCMC methods can be found in Gilks et al. (1996). BUGS (Spiegelhalter et al., 1996), WinBUGS software (Spiegelhalter et al., 2003) and JAGS (Plummer, 2003) use MCMC techniques to generate samples from posterior distribution of complicated models, providing an effective way to evaluate Bayesian models.

A rich literature now surrounds the theory and practice of MCMC methods. Review material can be found in Neal (1998), Smith and Roberts (1993), Tierney (1994), Besag et al. (1995), and Kass et al. (1998). Gelman et al. (2014) provides an excellent literature on MCMC methods. Further references on Bayesian computation appear in the books by Tanner (1993), Chen et al. (2000), and Robert and Casella (2004). Metropolis and Ulam (1949) and Metropolis et al. (1953) apparently were the first to describe Markov chain simulation of probability distributions (that is, the 'Metropolis algorithm'). Their algorithm was generalized by Hastings (1970). Chib and Greenberg (1995) for an elementary introduction and Tierney (1998) for a theoretical perspective. The conditions for Markov chain convergence appear in probability texts such as Feller (1968), and more recent work such as Rosenthal (1995) has evaluated the rates of convergence of Markov chain algorithms for statistical models. This section illustrates the use of Markov chain Monte Carlo (MCMC) algorithms in summarizing posterior distribution. MCMC algorithms are

attractive and are easy to set up a programm and requires relatively little prior input from the user. R is a convenient language for programming these algorithms and is also very suitable for programming output analysis. The details of these algorithm through R software will be discussed in Section 1.8. This section illustrates the various MCMC algorithm which are used throughout the thesis, first the algorithm of MCMC is summarized as

1.6.1 The algorithm

A Markov chain is a stochastic process $\{\theta^{(1)}, \theta^{(2)}, \ldots, \theta^{(T)}\}$ such that

$$p(\theta^{t+1}|\theta^{(t)}, \ldots, \theta^{(1)}) = p(\theta^{t+1}|\theta^{(t)});$$

that is, the distribution of θ at sequence $t+1$ given all the preceding θ values (for times $t, t-1, \ldots, 1$) depends only on the value $\theta^{(t)}$ of the previous sequence t. Moreover, $p(\theta^{t+1}|\theta^{(t)})$ is independent of time t . Finally, when the Markov chain is irreducible, aperiodic, and positive-recurrent, as $t \to \infty$ the distribution of $\theta^{(t)}$ converges to its equilibrium distribution, which is independent of the initial values of the chain $\theta^{(0)}$; Gilks et al. (1996).

In order to generate a sample from $(p(\theta|y))$, we must construct a Markov chain with two desired properties: (1) $p(\theta^{t+1}|\theta^{(t)})$ should be "easy to generate from," and (2) the equilibrium distribution of the selected Markov chain must be the posterior distribution of interest $(p(\theta|y))$, Ntzoufras (2009).

Assuming that we have constructed a Markov chain with these requirements, we then

1. Select an initial value $\theta^{(0)}$.
2. Generate T values until the equilibrium distribution is reached.
3. Monitor the converge of the algorithm using convergence diagnostics. If convergence diagnostics fail, we then generate more observations.
4. Cut off the first B observations.
5. Consider $\theta^{(B+1)}, \theta^{(B+2)}, \ldots, \theta^{(T)}$ as the sample for the posterior analysis.
6. Plot the posterior distribution (usually focus is on the univariate marginal distributions).

7. Finally, obtain summaries of the posterior distribution (mean, median, standard deviation, quantiles).

Various MCMC algorithms are available but the two most important algorithms are the Metropolis-Hastings algorithm (Metropolis et al., 1953; Hastings, 1970) and the Gibbs sampling (Geman and Geman, 1984). Some important more recent developments reported in the MCMC literature are the slice sampler (Higdon, 1998; Damien et al., 1999; Neal, 2003), the reversible jump MCMC (RJMCMC) algorithm (Green, 1995), and perfect sampling (Propp and Wilson, 1996; Merller, 1999), but our main focus is only on Meropolis algorithms. For additional information regarding MCMC specific methods have been discussed in Gilks et al. (1996), Robert and Casella (2004), Givens and Hoeting (2005), and Gamerman and Lopes (2006). The two most important MCMC algorithm would be discussed in detail in the following section.

1. Gibbs Sampling
2. Metropolis-Hasting algorithm

1.6.2 Gibbs sampling

One of the attractive methods for setting up an MCMC algorithm is Gibbs sampling. It was introduced by Geman and Geman (1984). One advantage of the Gibbs sampler is that, in each step, random values must be generated from unidimensional distributions for which a wide variety of computational tools exists (Gilks et al., 1996). Frequently, these conditional distributions have a known form and, thus, random numbers can be easily simulated using standard functions in statistical and computing software. Gibbs sampling is always moving to new values and, most importantly, does not require specification of proposal distributions. On the other hand, it can be ineffective when the parameter space is complicated or the parameters are highly correlated.

The Gibbs sampler is a special case of single-component Metropolis-Hastings algorithm using as proposal density $q(\theta'|\theta^{(t)})$ the full conditional posterior distribution $p(\theta_j|\theta_{j'}, y)$,where $\theta_{j'} = (\theta_1, \ldots, \theta_{j-1}, \theta_{j+1}, \ldots, \theta_d)^T$. The algorithm can be summarized by the following steps:

1. Set initial values $\theta^{(0)}$.

2. For $t = 1,\ldots,T$ repeat the following steps

   a. Set $\theta = \theta^{(t-1)}$

   b. For $j = 1,\ldots,d$, update θ_j from $\theta_j \sim p(\theta_j|\theta_{j'}, y)$.

   c. Set $\theta^{(t)} = \theta$ and save it as the generated set of values at $t+1$ iteration of the algorithm.

Hence, given a particular state of the chain $\theta^{(t)}$, we generate the new parameter values by

$$\begin{aligned}
&\theta_1^{(t)} \sim p(\theta_1|\theta_2^{(t-1)},\theta_3^{(t-1)},\ldots,\theta_p^{(t-1)},y),\\
&\theta_2^{(t)} \sim p(\theta_2|\theta_1^{(t)},\theta_3^{(t-1)},\ldots,\theta_p^{(t-1)},y),\\
&\theta_3^{(t)} \sim p(\theta_3|\theta_1^{(t)},\theta_2^{(t)},\theta_4^{(t-1)}\ldots,\theta_p^{(t-1)},y),\\
&\quad\vdots,\\
&\theta_j^{(t)} \sim p(\theta_j|\theta_1^{(t)},\theta_2^{(t)},\ldots,\theta_{j-1}^{(t)},\theta_{j+1}^{(t-1)},\theta_p^{(t-1)},y),\\
&\quad\vdots,\\
&\theta_p^{(t)} \sim p(\theta_p|\theta_1^{(t)},\theta_2^{(t)},\ldots,\theta_{p-1}^{(t)},y).
\end{aligned}$$

More detailed description of the Gibbs sampler is given by Casella and George (1992) and Smith and Roberts (1993), while early applications of the Gibbs sampling are provided by Gelfand and Smith (1990) and Gelfand et al. (1990). A gentle introduction can be found in Gelman et al. (2014).

1.6.3 Metropolis-Hastings algorithm

Metropolis et al. (1953) originally formulated the Metropolis algorithm, by introducing the Markov-chain-based simulation methods used in science. Later, Hastings (1970) generalized the original method in what is known as the Metropolis-Hastings algorithm. The latter is considered to be the general formulation of all MCMC methods. Green (1995) further generalized the Metropolis-Hastings algorithm by introducing reversible jump Metropolis- Hastings algorithms for sampling from parameter spaces with different dimensions.

Let as assume a target distribution $p(\theta|y)$ from which we wish to generate a sample of size T . The Metropolis-Hastings algorithm can be described by the following iterative steps; where $\theta^{(t)}$ is the vector of generated values in t iteration of the algorithm:

1. Set initial values $\theta^{(0)}$.

2. For $t = 1, \ldots, T$ repeat the following steps

   a. Set $\theta = \theta^{(t-1)}$

   b. Generate new candidate values θ' from a proposal distribution $q(\theta'|\theta)$.

   c. Calculate
   $$\alpha = min\left(1, \frac{p(\theta'|y)q(\theta|\theta')}{p(\theta|y)q(\theta'|\theta)}\right)$$

   d. Update $\theta^{(t)} = \theta'$ with probability α; otherwise $\theta^{(t)} = \theta$.

The Metropolis-Hastings algorithm will converge to its equilibrium distribution regardless of whatever proposal distribution q is selected. Nevertheless, in practice, choice of the proposal is important since poor choices will considerably delay convergence towards the equilibrium distribution. Special cases of Metropolis-Hastings algorithm are random-walk Metropolis algorithm, Independence sampler or independent Metropolis algorithm. These commonly used algorithm adaptations are described below.

1.6.3.1 Random walk Metropolis algorithm

Random walk Metropolis (RWM) algorithms are widely used generic Markov chain Monte Carlo (MCMC) algorithms. The ease with which RWM algorithms can be constructed has no doubt played a pivotal role in their popularity. The efficiency of a RWM algorithm depends fundamentally upon the scaling of the proposal density. Choose the variance of the proposal to be too small and the RWM will converge slowly since all its increments are small. Conversely, choose the variance of the proposal to be too large and too high a proportion of proposed moves will be rejected. Of particular interest is how the scaling of the proposal variance depends upon the dimensionality of the target distribution. The target distribution is the distribution of interest and the MCMC algorithm is constructed such that the stationary distribution of the Markov chain is the target distribution. Work in which the random walk Metropolis algorithm is used as part of the computational scheme includes Muller and Rios Insua (1995), Sargent (1997, 1998), Kuo and Yang (1996), Greenhouse and Wasserman (1996), Newton et al. (1996), Verdinelli and Wasserman (1998), Muller and Roeder (1997) and Waller et al. (1997). The popularity of this approach is probably due to the ease with which the algorithm is implemented. The algorithm of random walk Metropolis can be summarized by;

1. Specify initial value $\theta^{(0)}$

2. For $t = 1, \ldots, T$

   - Set $\theta = \theta^{(t-1)}$
   - Propose a new value θ' from $N(\theta, \bar{s}_\theta^2)$
   - Calculate $\log \alpha = \min(0, A)$ with A given by

$$A = \log \frac{p(y|\theta')p(\theta')}{p(y|\theta)p(\theta)} = (\theta' - \theta)\left(y - \frac{\theta' + \theta - 2\mu_\theta}{2\sigma_\theta^2}\right) - N \log \frac{1 + e^{\theta'}}{1 + e^{\theta}}$$

3. Set $\theta^{(t)} = \theta'$ with probability α and $\theta^{(t)} = \theta$ with the remaining probability.

Parameter $\bar{s}_\theta^2$ is a tunnig parameter that need to be calibrated such that it achieves an acceptance rate approximately equal to 25%. The optimal acceptance rate according to Roberts et al. (1997) and Neal and Roberts (2008) is around 25%, ranging from 0.23 for large dimensions to 0.45 for the univariate case Roberts and Rosenthal (2001).

The framework in which we have worked is that we obtain posterior mode and posterior variance by making use of `LaplaceApproximation`. For normal proposal, mean of the normal is set to posterior mode whereas variance of the proposal density is inflated by a factor $2.4^2/d$. Where, d is dimension of θ, (Gelman et al., 2014 p-290).

1.6.3.2 Independent Metropolis algorithm

The iterative steps of independent Metropolis algorithm can be described as follows

1. For $t = 1, \ldots, T$

   - Set $\theta = \theta^{(t-1)}$
   - Propose a new value θ' from $N(\bar{\mu}_\theta, \bar{s}^2)$
   - Calculate $\log \alpha = \min(0, A)$ with A given by

$$A = \log \frac{p(y|\theta')p(\theta')}{p(y|\theta)p(\theta)} + \log \frac{p_N(\theta; \bar{\mu}_\theta, \bar{s}_\theta^2)}{p_N(\theta'; \bar{\mu}_\theta, \bar{s}_\theta^2)} \tag{1.16}$$

$$= (\theta' - \theta)\left(y - \frac{\theta' + \theta - 2\mu_\theta}{2\sigma_\theta^2} + \frac{\theta' + \theta - 2\bar{\mu}_\theta}{2\bar{s}_\theta^2}\right) - N \log \frac{1 + e^{\theta'}}{1 + e^{\theta}}$$

2. Set $\theta^{(t)} = \theta'$ with probability α and $\theta^{(t)} = \theta$ with the remaining probability.

1.7 Model comparison

Model comparison is a crucial part of any statistical analysis. Due to recent computational advances, sophisticated techniques for Bayesian model comparison in survival analysis are becoming increasingly popular. There has been a resent surge in the statistical literatureon Bayesian methods for model comparison, including articles by George and McCulloch (1993), Ibrahim and Laud (1994), Kass and Raftery (1995). Articles focusing on Bayesian approaches to model comparison in the context of survival analysis include Madigan and Raftery (1994), Ibrahim and Chen (1998) and Ibrahim et al. (1999).

No statistical analysis is complete without testing the adequacy of the model upon which the analysis is based. In this section, we describe the most popular and efficient goodness of fit criterion that can be applied to most of the models described in the subsequent chapters.

In Bayesian inference, the most common method of assessing the goodness of fit of an estimated statistical model is a generalization of the frequentist Akaike Information Criterion (AIC). The Bayesian method, like AIC, is not a test of the model in the sense of hypothesis testing, though Bayesian inference has Bayes factors for such purposes. Instead, like AIC, Bayesian inference provides a model fit statistic that is to be used as a tool to refine the current model or select the better-fitting model of different methodologies. To begin with, model fit can be summarized with deviance, which is defined as -2 times the log-likelihood (Gelman et al. 2004, p. 180), such as

$$D(y,\theta) = -2log[p(y|\theta)]$$

Just as with the likelihood, $p(y|\theta)$, or log-likelihood, the deviance exists at both the record and model-level. It is possible to have a negative deviance. Deviance is derived from the likelihood, which is derived from probability density functions (PDF). Evaluated at a certain point in parameter space, a PDF can have a density larger than 1 due to a small standard deviation or lack of variation. Likelihoods greater than 1 lead to negative deviance, and are appropriate.

The deviance information criterion (DIC) (Spiegelhalter et al. 2002) is the second model assessment tool used in this thesis, and it is a Bayesian alternative to Akaike's information criterion (AIC) and the Bayesian information criterion (BIC, also known as the Schwarz criterion). The DIC uses the posterior densities, which means that it takes the prior information into account. The criterion can be applied

to nonnested models and models that have non-iid data. Calculation of the DIC in MCMC is trivial-it does not require maximization over the parameter space, like the AIC and BIC. A smaller DIC indicates a better fit to the data set.

Letting θ be the parameters of the model, the deviance information formula is

$$DIC = D(\bar{\theta}) + pD = D(\bar{\theta}) + 2pD \tag{1.17}$$

where, $D(\theta)$ is the deviance and pD is effective number of parameters.

1.8 The statistical software

To fit the Bayesian survival models, one needs a statistical computing environment. An environment that meets these requirements is the `R`, R Core Team, (2015) software. The open source R statistical computing environment provides sufficient packages and functions to elaborate the importance of Bayesian theory. It gives all necessary information about the data which an analyst is requires. The tools and techniques which are used within Bayesian framework are implemented in `LaplacesDemon` Statisticat (2015) package. Traditionally, it has been difficult to develop closed-form expressions for posterior distribution, except in the simplest cases. However, with the advent of MCMC methods, Bayesian methods are being easily implemented. MCMC methods are computer-intensive methods that allow one to simulate draws from the posterior distribution, without having to calculate the posterior distribution. The goal of `LaplacesDemon` is to provide a complete and self-contained Bayesian environment within `R`. Currently, we are using stable version of `LaplacesDemon_15.03.19`, which has a lot of new implementations. It requires only 2.6 MB space. Its source code can be obtained from us. The old version of `LaplacesDemon_13.03.04` is available on the cran which are downloaded from

`https://cran.r-project.org/src/contrib/Archive/LaplacesDemon/.`

Open R, and install the LaplacesDemon package from source.

```
>install.packages(pkgs="path/LaplacesDemon_ver.tar.gz",repos=NULL,
 type="source")
```

where `path` is a path to the zipped source code, and `ver` is replaced with the latest version found in the name of the downloaded file. Once installed, simply use the

library or require function in R to activate the LaplacesDemon package and load its functions into memory.

```
> library(LaplacesDemon)
```

The two main functions which are used to estimate the parameters of the models are LaplaceApproximation which gives the approximated posterior results and then give the simulated summary by sampling importance resampling, and LaplacesDemon function which gives the simulated posterior summary. Currently, 19 methods of optimization are available with LaplaceApproximation. However, we have found that trust region, Nelder-Mead and BFGS are better than the others. For the purpose of simulation, currently, 41 algorithm of simulations are available with LaplacesDemon. However, we have preferred to use random walk Metropolis and independent Metropolis algorithms. The Laplace method is a family of asymptotic techniques used to approximate integrals. Since its introduction, the Laplace approximation has been applied successfully in many disciplines. In the 1980, the Laplace approximation experienced renewed interest, especially in statistics. Lindley (1980) has proposed approximations for the moments that capture the first-order error terms of normal approximation. However, it required evaluation of third derivatives of posterior, a difficult task to implement in many practical situations. Some improvements in its implementation were introduced (Tierney and Kadane, 1986; Tierney et al., 1989). Only since the 1980 has the Laplace approximation been seriously considered by statisticians in practical applications. Bernardo and Smith (2000) note that Laplace approximation is an attractive numerical approximation algorithm, and will continue to develop.
Arguments and details of LaplaceApproximation and LaplacesDemon are as:

1.8.1 LaplaceApproximation

The arguments of LaplaceApproximation is given in the following:

```
LaplaceApproximation(Model,parm,Data,Interval=1e-06,
Iterations= 1000,Method ="SPG",Samples=1000,sir=TRUE,
Stop.Tolerance =1e-05)
```

where Model receives the model from a user-defined function. This function passes two arguments to the model function, parm and Data. The argument

parm is a vector of initial values equal in length to the number of parameters. LaplaceApproximation will attempt to optimize these initial values for the parameters, where the optimized values are the posterior modes, for later use with the LaplacesDemon. Data argument accepts a list of data. The list of data must include mon.names which contains monitored variable names, and parm.names which contains parameter names. LaplaceApproximation must be able to determine the sample size of the data, and will look for a scalar sample size variable n or N. If not found, it will look for variable y or Y, and attempts to take its number of rows as sample size. LaplaceApproximation needs to determine sample size due to the asymptotic nature of this method. Sample size should be at least sqrt(J) with J exchangeable parameters. The argument Iterations refers to a cycle of the algorithm that generates a full set of parameter values from the posterior distribution. It is frequently used to denote an observation of simulated values. This argument accepts an integer that determines the number of iterations that LaplaceApproximation will attempt to maximize the logarithm of the unnormalized joint posterior density. This package have several optimization method such as AGA, HAR (hit and run), Rprop. The default method is this function is SPG. These methods of optimization can be used through the argument Method. The two optimization methods used in this thesis in most of the analysis are Nelder-Mead and trust region method i.e Method='NM' or 'TR'. The details of these methods are in the next section. The argument Samples indicates the number of posterior samples to be taken with sampling importance resampling via the SIR function, which occurs only when sir=TRUE. Note that the number of samples should increase with the number and intercorrelations of the parameters. sir indicates whether or not sampling importance resampling (SIR) is conducted via the SIR function to draw independent posterior samples. This argument defaults to TRUE. Even when TRUE, posterior samples are drawn only when LaplaceApproximation has converged.

The speed of LaplaceApproximation depends on the optimization algorithm selected, and typically involves many evaluations of the objective function per iteration (where an MCMC algorithm with a multivariate proposal usually evaluates once per iteration), making many MCMC algorithms faster per iteration. The attractiveness of LaplaceApproximation is that it typically improves the objective function better than iterative quadrature and MCMC when the parameters are in low-probability regions. LaplaceApproximation is also typically faster than MCMC because it is seeking point-estimates, rather than attempting to represent the target distribution with enough simulation draws. LaplaceApproximation extends MLE, but shares similar limitations, such as its asymptotic nature with

respect to sample size and that marginal posterior distributions are Gaussian.

1.8.2 Details of optimization method

1. **Nelder and Mead:** When `Method`="`NM`", the Nelder-Mead (1965) algorithm is used. Nelder-Mead is a derivative-free, direct search method that is known to become inefficient in large-dimensional problems. As the dimension increases, the search direction becomes increasingly orthogonal to the steepest ascent (usually descent) direction. However, in smaller (10-20) dimensions, it is a popular algorithm.

2. **BFGS**: When `Method`="`BFGS`", the BFGS method is used, which was proposed by Broyden (1970), Fletcher (1970), Goldfarb (1970), and Shanno (1970), independently. BFGS may be the most efficient and popular quasi-Newton optimiziation algorithm.

3. **Newton-Raphson:** When `Method`="`NR`", the Newton-Raphson optimization algorithm, also known as Newton's Method, is used. Although this method is common among the Mathematicians but it is not ideal choice in statistical optimization problem. The main reason behind is the requirement of guess values close to the optimal,which is a difficult task in modelling. Moreover, it requires derivatives and inverse of the Hessian matrix which can be computationally expensive.

4. **The trust region:** When `Method`="`TR`", The trust region algorithm of Nocedal and Wright (1999) is used. The TR algorithm attempts to reach its objective in the fewest number of iterations, is therefore very efficient, as well as safe. The efficiency of TR is attractive when model evaluations are expensive. The Hessian is approximated each iteration, making TR best suited to models with small to medium dimensions, say up to a few hundred parameters.

After `LaplaceApproximation` finishes, due either to early convergence or completing the number of specified iterations, it approximates the Hessian matrix of second derivatives, and attempts to calculate the covariance matrix by taking the inverse of the negative of this matrix.

Out of these optimization algorithm Nelder-Mead and trust region methods are used in the thesis. These methods are implemented on real survival data and its all necessary posterior summaries are illustrated in detail.

1.8.3 LaplacesDemon

The arguments of `LaplacesDemon` is given in the following:

```
LaplacesDemon(Model,Data, Initial.Values,Covar=NULL,
Iterations =1e+05,Status = 1000, Thinning = 100,
Algorithm = "MWG")
```

The function `Model` required argument receives the model from a user-defined function. The user-defined function is where the model is specified. `LaplacesDemon` passes two arguments to the model function, `parm` and `Data`, and receives five arguments from the model function: `LP` (the logarithm of the unnormalized joint posterior), `Dev` (the deviance), `Monitor` (the monitored variables), `yhat` (the variables for posterior predictive checks), and `parm`, the vector of parameters, which may be constrained in the model function.

`Data` argument accepts a list of data. The list of data must contain `mon.names` which contains monitored variable names, and must contain `parm.names` which contains parameter names. The `as.parm.names` function may be helpful for preparing the data.

For `LaplacesDemon`, `Initial.Values` argument requires a vector of initial values equal in length to the number of parameters. Initial value will be the starting point for an adaptive chain or a non-adaptive Markov chain of a parameter. If all initial values are set to zero, then Laplace's Demon will attempt to optimize the initial values with the `LaplaceApproximation` function. After Laplace's Demon finishes updating, it may be desired to continue updating from where it left off.

The argument `Iteration` accepts integers larger than **10**, and determines the number of iterations that Laplace's Demon will update the parameters while searching for target distributions.

1.8.4 Details of simulation algorithms

The `LaplacesDemon` offers numerous MCMC algorithms for simulation in Bayesian inference, and are, random walk Metropolis, Metropolis within Gibbs, independent Metropolis, delayed rejection Metropolis and many more.

1.8.4.1 Random-walk Metropolis

In the original Metropolis algorithm (Metropolis et al., 1953), only symmetric proposals of type $q(\theta'|\theta) = q(\theta|\theta')$ were considered. Random-walk Metropolis is a special case with $q(\theta'|\theta) = q(|\theta' - \theta|)$. Both cases result in an acceptance probability that depends only on the posterior (target) distribution

$$\alpha = \min\left(1, \frac{p(\theta'|y)}{p(\theta|y)}\right) = \min\left(1, \frac{p(y|\theta')p(\theta')}{p(y|\theta)p(\theta)}\right).$$

Random walk Metropolis algorithm is used in `LaplacesDemon` function as `Algorithm="RWM"` as given below:

```
LaplacesDemon(Model, Data=MyData, Initial.Values,
Covar=NULL, Iterations=1000, Status=100, Thinning=1,
Algorithm="RWM", Specs=NULL)
```

1.8.4.2 Independence Metropolis algorithm

Proposed by Hastings (1970) and popularized by Tierney (1994), the independence Metropolis `(IM)` algorithm (also called the independence sampler) is an algorithm in which the proposal distribution does not depend on the previous state or iteration. The proposal distribution must be a good approximation of the target distribution for the `IM` algorithm to perform well, and the proposal distribution should have slightly heavier tails than the target distribution. `IM` is used most often to obtain additional posterior samples given an algorithm that has already converged. Since `IM` is non-adaptive and uses a proposal distribution that remains fixed for all iterations, it may be used as a final algorithm. Also as IM algorithm needs close approximation which can be obtained from function `LaplaceApproximation`, then this algorithm will be implemented in `LaplacesDemon` function through the argument `Algorithm="IM"`, hence is an efficient algorithm. The command for the implementation of IM algorithm could be seen in the following:

```
LaplacesDemon(Model, Data=MyData, Initial.Values,
Covar=Fit$Covar, Iterations=1000, Status=100, Thinning=1,
Algorithm="IM",Specs=list(mu=Fit$Summary1[1:length(Initial.Values),1]))
```

1.8.4.3 Metropolis within Gibbs

Metropolis-within-Gibbs `(MWG)` is the original MCMC algorithm, introduced in Metropolis et al. (1953). Since it was the original MCMC algorithm, it pre-dated

Gibbs sampling (Gibbs), and was not known as Metropolis-within-Gibbs. MWG was later proposed as a hybrid algorithm that combines Metropolis-Hastings and Gibbs sampling, and was suggested in Tierney (1994). The idea was to substitute a Metropolis step when Gibbs sampling fails. `MWG` is a componentwise algorithm, meaning that each parameter is updated individually each iteration. This implies that the model specification function is evaluated a number of times equal to the number of parameters, per iteration. A componentwise proposal is generated randomly and the model is evaluated with the proposed parameter. If the proposal is an improvement in the logarithm of the unnormalized joint posterior density, then the proposal is accepted. If the proposal is not an improvement, then the proposal is accepted or rejected according to a probability. Since MWG is a componentwise algorithm, it is most efficient when the acceptance rate of each parameter is **0.44**. The advantage of MWG over the multivariate version, RWM, is that it is more efficient with information per iteration, so convergence is faster in iterations. The disadvantages of MWG are that covariance is not included in proposals, and it is more time-consuming due to the evaluation of the model specification function for each parameter per iteration.

```
LaplacesDemon(Model, Data=MyData, Initial.Values,
Covar=NULL, Iterations=1000, Status=100, Thinning=1,
Algorithm="MWG", Specs=NULL)
```

Independence Metropolis and random walk Metropolis algorithms are used for simulation throughout the thesis. These simulation tools has been practically implemented on survival data.

1.9 Conclusion

A pragmatic rationale for the use of Bayesian methods is the inherent flexibility introduced by their incorporation of multiple levels of randomness and the resultant ability to combine information from different sources, while incorporating all reasonable sources of uncertainty in inferential summaries. Such methods naturally lead to smoothed estimates in complicated data structures and consequently have the ability to obtain better real-world answers.

Another reason for focusing on Bayesian methods is more psychological, and involves the relationship between the statistician and the client or specialist in the subject matter area who is the consumer of the statistician's work. In many

practical cases, clients will interpret interval estimates provided by statisticians as Bayesian intervals, that is, as probability statements about the likely values of unknown quantities conditional on the evidence in the data. Such direct probability statements require prior probability specifications for unknown quantities (or more generally, probability models for vectors of unknowns), and thus the kinds of answers clients will assume are being provided by statisticians, Bayesian answers, require full probability models-explicit or implicit.

Finally, Bayesian inferences are conditional on probability models that invariably contain approximations in their attempt to represent complicated real-world relationships. If the Bayesian answers vary dramatically over a range of scientifically reasonable assumptions that are unassailable by the data, then the resultant range of possible conclusions must be entertained as legitimate, and we believe that the statistician has the responsibility to make the client aware of this fact, (Gelman et al. 2014).

In this thesis, we focus on the construction and modelling of real survival data in a complete Bayesian environment. We have written `R` functions for some complex models such as exponentiated Weibull, Lomax, Weibull Lomax and exponential Lomax distributions to draw Bayesian inference and also for the purpose of simulation.

CHAPTER 2

Generalized Exponential Model: A Bayesian Study

2.1 Introduction

The exponential distribution occupies an important position in lifetime distribution study. Historically, the exponential distribution was the first lifetime model for which statistical methods were extensively developed. Early work by Sukhatme (1937) and later work by Epstein and Sobel (1953, 1954, 1955) and Epstein (1954, 1960) gave numerous results and popularized the exponential as a lifetime distribution. Gupta and Kundu (2001), presented the generalized exponential distribution. The generalized exponential (GE) distribution has lots of interesting properties and it can be used quite effectively to analyze several skewed life time data. Since the distribution function of the GE is in closed form, it can be used quite easily for analyzing censored data also. This family has lots of properties which are quite similar to those of a gamma distribution but it has an explicit expression of the survival function like a Weibull distribution. Gupta and Kundu (2007) provided a detailed review and some developments on the generalized exponential distribution. In this chapter, we consider a two-parameter extension of exponential distribution. The two parameters of the GE distribution represents the shape and scale parameter. It is observed that the GE family always has a decreasing probability function like an exponential distribution but it allows for increasing, decreasing and constant hazard rates like a Weibull distribution or an exponentiated exponential distribution. The GE distribution has an explicit expression of survival function and failure rate hazard function.

Due to convenient form of the distribution function, simulation can easily be made from GE distribution. This chapter covers the Bayesian inference procedures for exponential distribution. The two-parameter extension of exponential distribution is a particular member of the three-parameter generalized power Weibull distribution, introduced by Nikulin and Haghighi (2006). Moreover, the GE distribution is a special case of Gurvich (1977). This chapter deals with two forms of exponential distribution, first is generalized exponential and second is exponential extension distribution. The GE distribution has the distribution function

$$F(t;\alpha,\lambda) = \left(1 - exp\left\{-\frac{t}{\lambda}\right\}\right)^{\alpha};\ \alpha,\lambda,t > 0 \tag{2.1}$$

Therefore, the pdf of GE distribution

$$f(t;\alpha,\lambda) = \frac{\alpha}{\lambda}\left(1 - exp\left\{-\frac{t}{\lambda}\right\}\right)^{\alpha-1} exp\left\{-\frac{t}{\lambda}\right\} \tag{2.2}$$

Gupta and Kundu (1999) provided the graphs of the generalized exponential density func- tions for different values of shape parameter. The density functions of the generalized exponential distribution can take different shapes. For $\alpha \leq 1$, it is a decreasing function and for $\alpha > 1$, it is a unimodal, skewed, right tailed similar to the Weibull or gamma density function, reported in Figure 2.1. It is observed that even for very large shape parameter, it is not symmetric. For $\alpha = 1$, the mode is at $\log\alpha$ for $\alpha > 1$ and for $\alpha < 1$, the mode is at 0. The mean, median and mode are non-linear functions of the shape parameter and as the shape parameter goes to infinity all of them tend to infinity. For large values of α, the mean, median and mode are approximately equal to $\log\alpha$ but they converge at different rates.
The survival function of GE distribution,

$$S(t;\alpha,\lambda) = 1 - \left(1 - exp\left\{-\frac{t}{\lambda}\right\}\right)^{\alpha} \tag{2.3}$$

and hazard function,

$$h(t;\alpha,\lambda) = \frac{\frac{\alpha}{\lambda}\left(1 - exp\left\{-\frac{t}{\lambda}\right\}\right)^{\alpha-1} exp\left\{-\frac{t}{\lambda}\right\}}{\left(1 - exp\left\{-\frac{t}{\lambda}\right\}\right)^{\alpha}} \tag{2.4}$$

Due to the convenient structure of the GE distribution, it can be used quite effectively in analyzing many lifetime data. It is observed that the hazard function of the GE distribution can be increasing, decreasing or constant depending on the shape parameter α. For any λ, the hazard function is nondecreasing if $\alpha > 1$, it is decreasing if $\alpha < 1$ and for $\alpha = 1$, it is constant. The plots of the hazard functions for different values of α can be obtained as in Gupta and Kundu (1999) and in this chapter they are presented in Figure 2.1. GE distribution with the shape parameter α and the scale parameter λ will be denoted by GE(α, λ). GE(1, λ) represents the exponential distribution with the scale parameter λ.

2.1.1 Functions for generalized exponential distribution in R

1. R code for probability density function is

```
dgenexp<-function(x,alpha,lambda){
d1<-alpha*dexp(x,1/lambda)
d2<-pexp(x,1/lambda)^(alpha-1)
d<-(d1*d2)
return(d)
}
```

2. R code for cumulative density function is

```
pgenexp<-function(x,alpha,lambda){
p<-pexp(x,1/lambda)^alpha
return(p)
}
```

3. R code for random generation function is

```
rgenexp<-function(n,shape,scale)
{
u<-runif(n)
x<--scale*log(1-u^(1/shape))
return(x)
}
```

4. R code for survival function is

```
sgenexp<-function(x,alpha,lambda){
surv<-(1-pgenexp(x,alpha,lambda))
return(surv)
}
```

5. R code for hazard function is

```
hgenexp<-function(x,alpha,lambda){
haz<-dgenexp(x,alpha,lambda)/sgenexp(x,alpha,lambda)
return(haz)
}
```

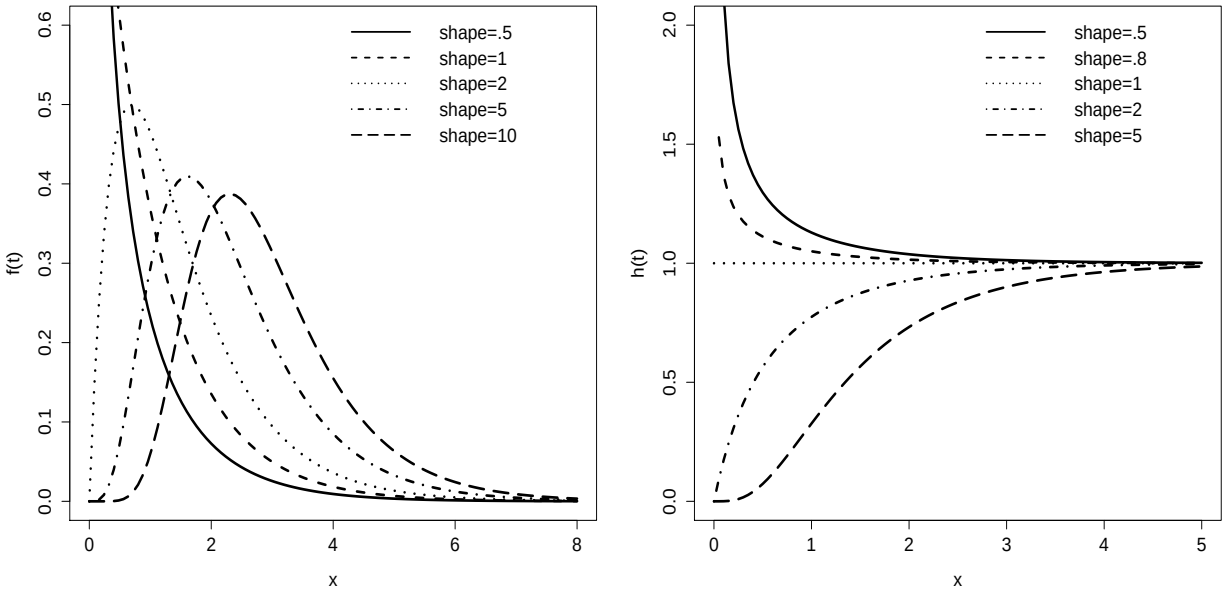

Figure 2.1: *pdf and hazard curves of GE distribution with different values of shape and at scale is equal to one.*

The second distribution which will analyzed in Bayesian framework is exponential extension (EE) distribution having probability density function, survival and hazard function are,

$$f(t;\alpha,\lambda) = \frac{\alpha}{\lambda}\left(1+\frac{t}{\lambda}\right)^{\alpha-1} exp\left(1-\left(1+\frac{t}{\lambda}\right)^{\alpha}\right) \tag{2.5}$$

$$S(t;\alpha,\lambda) = exp\left(1-\left(1+\frac{t}{\lambda}\right)^{\alpha}\right) \tag{2.6}$$

$$h(t;\alpha,\lambda) = \frac{\alpha}{\lambda}\left(1+\frac{t}{\lambda}\right)^{\alpha-1} \tag{2.7}$$

respectively.

2.1.2 Functions for exponential extension distribution in R

1. R code for probability density function is

```
dexpext<-function(y,shape,scale){
d1<-shape/scale*(1+y/scale)^(shape-1)
d2<-exp(1-(1+y/scale)^shape)
```

```
d<-d1*d2
return(d)
}
```

2. R code for cumulative density function is

```
pexpext<-function(y,shape,scale) 1- exp(1-(1+y/scale)^shape)
```

3. R code for random generation function is

```
rexpext<-function(n,shape,scale){
u<-runif(n)
t<--scale*((log(1-u)-1)^(1/shape)-1)
return(t)
}
```

4. R code for survival function is

```
surv<-function(y,shape,scale) exp(1-(1+y/scale)^shape)
```

5. R code for hazard function is

```
haz<-function(y,shape,scale) shape/scale*(1+y/scale)^(shape-1)
```

Both GE and EE have exponential distribution with one parameter at $\alpha = 1$. The aim of this chapter is to implement Bayesian analytic and simulation tools to fit generalized exponential distribution (having parameters shape and scale), to study survival data in Bayesian scenario. Model comparison will be discussed in Section 2.7

2.2 Regression model

An important way of handling heterogeneity in a population is through the inclusion of regressor variables in the model. It is very common for data to involve regressor variables related to lifetime: for example, in a study on survival time for lung cancer patients, factors such as the age and general physical condition of the patient, the type of tumour, the time since diagnosis, and so on may all be relevant. Regression models, with lifetime as the response variable and the concomitant variables as regressor variables, allow such additional factors to be conveniently incorporated in a statistical analysis.

2.2.1 Generalized exponential regression model

We first generalize the GE distribution. Recall that for GE distribution the hazard function is $h(t;\alpha,\lambda) = \frac{\alpha/\lambda(1-e^{-t/\lambda})^{\alpha-1}e^{-t/\lambda}}{1-(1-e^{-t/\lambda})^{\alpha}}$ with respect to time. Model the hazard as a function of the covariate vector x, Tablemann and Kim (2004).

Assume the hazard function at time t for an individual has the form

$$h(t|x) = h_0(t).e^{x^T\beta} \tag{2.8}$$

Therefore, for GE distribution,

$$h(t|x) = \frac{\alpha/\lambda(1-e^{-t/\lambda})^{\alpha-1}e^{-t/\lambda}}{1-(1-e^{-t/\lambda})^{\alpha}}.e^{x^T\beta} \tag{2.9}$$

Consequently,

$$h(t|x) = \frac{\alpha/\lambda(1-e^{-t/\lambda})^{\alpha-1}e^{-t/\lambda}}{1-(1-e^{-t/\lambda})^{\alpha}}.e^{x_1^T\beta_1+\ldots+x_p^T\beta_p} \tag{2.10}$$

where, $\beta = [\beta_1, \beta_2, \ldots, \beta_p]$ is a vector of regression parameters. the function $h_0(t)$ is called the baseline hazard. It is the value of the hazard function when the covariate vector $x = 0$ or $\beta = 0$. The Equation 2.10 says that the covariates act multiplicatively on the hazard rate.

The survival function of T given x is,

$$S(t|x) = exp(-h(t|x)t) = exp\left(-\frac{\alpha/\lambda(1-e^{-t/\lambda})^{\alpha-1}e^{-t/\lambda}}{1-(1-e^{-t/\lambda})^{\alpha}}e^{x^T\beta}t\right). \tag{2.11}$$

Thus, the p.d.f of T given x is

$$f(t|x) = h(t|x)S(t|x)$$

$$f(t|x) = \frac{\alpha/\lambda(1-e^{-t/\lambda})^{\alpha-1}e^{-t/\lambda}}{1-(1-e^{-t/\lambda})^{\alpha}}.e^{x^T\beta}exp\left(-\frac{\alpha/\lambda(1-e^{-t/\lambda})^{\alpha-1}e^{-t/\lambda}}{1-(1-e^{-t/\lambda})^{\alpha}}e^{x^T\beta}t\right). \tag{2.12}$$

One important feature of survival data is the presence of censoring, which creates special problems in the analysis. Lifetime data are censored when the exact failure time for a specific trial is unknown. When analyzing censored data, Bayesian methods have an important advantage over classical methods. From a classical perspective, confidence interval and other inferential statements must be made with respect to repeated sampling of the data. An advantage of the Bayesian approach is that only the censoring pattern, e.g., a right-censored failure time, is relevant, not which censoring scheme, such as Type I, Type II, or random censoring, produced it. Likelihood function for right censored data will be discussed in the next section.

2.2.1.1 Construction of likelihood function of GE regression model with censoring.

Suppose that there are n subjects under study, and that associated with the ith individual is a survival time t_i and a censoring time t_{c_i}. The t's are assumed to be independent and identically distributed with density $f(t)$ and survival function $S(t)$. The exact survival time t_i of an individual will be observed only if $t_i \leq t_{c_i}$. The data in this framework can be represented by the n pairs of random variables (y_i, δ_i), where

$$y_i = min(t_i, t_{c_i})$$

and

$$\delta_i = \begin{cases} 1 & \text{if } t_i \leq t_{c_i}, \\ 0 & \text{if } t_i > t_{c_i}. \end{cases} \tag{2.13}$$

Then the likelihood function for $(\beta, h_0(.))$ for a set of right censored data on n subjects is given by

$$L \propto \prod_{i=1}^{n} f(y_i|x_i)^{\delta_i} S(t_{c_i}|x_i)^{1-\delta_i} \tag{2.14}$$

$$L = \prod_{i=1}^{n} \left[\left\{ \frac{\alpha/\lambda(1-e^{-y_i/\lambda})^{\alpha-1}e^{-y_i/\lambda}}{1-(1-e^{-y_i/\lambda})^{\alpha}} . e^{x_i^T\beta} exp\left(-\frac{\alpha/\lambda(1-e^{-y_i/\lambda})^{\alpha-1}e^{-y_i/\lambda}}{1-(1-e^{-y_i/\lambda})^{\alpha}} e^{x_i^T\beta} y_i \right) \right\}^{\delta_i} \right.$$
$$\left. \left\{ exp\left(-\frac{\alpha/\lambda(1-e^{-t_{c_i}/\lambda})^{\alpha-1}e^{-t_{c_i}/\lambda}}{1-(1-e^{-t_{c_i}/\lambda})^{\alpha}} e^{x_i^T\beta} t_{c_i} \right) \right\}^{1-\delta_i} \right] . \left\{ \frac{2\gamma}{\pi(\alpha^2+\gamma^2)} \right\} .$$
$$\prod_{j=1}^{p} \left\{ \frac{1}{\sqrt{2\pi}1000} e^{\frac{-\beta_j^2}{2*1000^2}} \right\} \tag{2.15}$$

2.2.1.2 Prior

Now, we set prior to the GE proportional hazard model whose likelihood function has been discussed above. If $Y \sim GE(\alpha, \lambda)$. Prior probabilities are specified for α and β:

$$\alpha \sim \text{half-Cauchy}(\gamma)$$
$$p(\alpha|\gamma) = \frac{2\gamma}{\pi(\alpha^2+\gamma^2)}, \quad \alpha > 0$$

The half-Cauchy distribution with scale parameter $\gamma = 25$ is used as a noninformative prior distribution for shape parameter. As Gelman and Hill (2007) recommend that, the uniform, or if more information is necessary the half-Cauchy is a better choice. For $\gamma = 25$, the half-Cauchy distribution becomes almost flat as it is evident from figure in the left panel of Figure 2.2.

Since, $\lambda > 0$ and β can take any value on the real line, hence, log link function is used

$$log(\lambda) = \mathbf{X}\beta$$

where, $\boldsymbol{X}$ is model matrix and $\boldsymbol{\beta}$ is the vector of regression coefficients,
or, equivalently,

$$\lambda = e^{\mathbf{X}\beta}$$

Each component of the β parameters is assigned a weak informative Gaussian prior probability distribution. Assuming that $\beta_i^{'}$s are independently distributed as normal with mean$= 0$ and standard deviation $= 1000$, so that a flat prior can be obtained. This is evident from figure in the right panel of Figure 2.2. The large variance indicates a lot of uncertainty about each β, and is hence a weak informative distribution

$$\beta_j \sim N(0, 1000)$$

. After assuming half-Cauchy distribution as a prior for shape parameter and

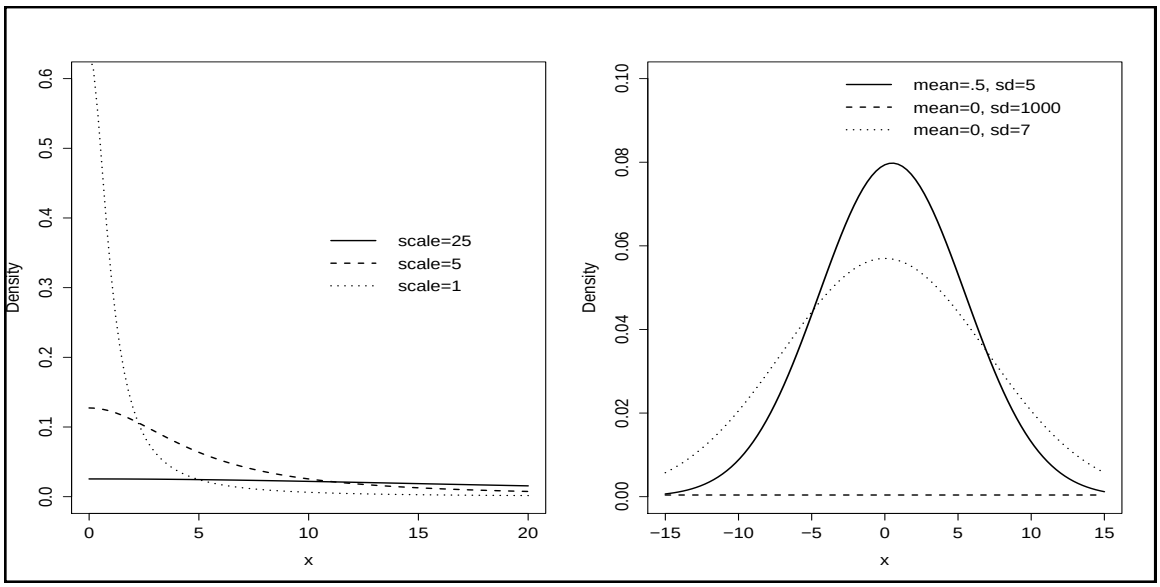

Figure 2.2: Density plots of half-Cauchy and normal prior distribution

normal distribution for β with mean 0 and sd 1000, the joint posterior distribution are

$$p(\beta, \alpha | y, X) = p(y | \alpha, \beta, X).p(\alpha).p(\beta) \tag{2.16}$$

$$p(\beta,\alpha|y) \propto \prod_{i=1}^{n}\left[\left\{\frac{\alpha/\lambda(1-e^{-y_i/\lambda})^{\alpha-1}e^{-y_i/\lambda}}{1-(1-e^{-y_i/\lambda})^{\alpha}}.e^{x_i^T\beta}exp\left(-\frac{\alpha/\lambda(1-e^{-y_i/\lambda})^{\alpha-1}e^{-y_i/\lambda}}{1-(1-e^{-y_i/\lambda})^{\alpha}}e^{x_i^T\beta}y_i\right)\right\}^{\delta_i}\right.$$
$$\left.\left\{exp\left(-\frac{\alpha/\lambda(1-e^{-t_{c_i}/\lambda})^{\alpha-1}e^{-t_{c_i}/\lambda}}{1-(1-e^{-t_{c_i}/\lambda})^{\alpha}}e^{x_i^T\beta}t_{c_i}\right)\right\}^{1-\delta_i}\right].\left\{\frac{2\gamma}{\pi(\alpha^2+\gamma^2)}\right\}.$$

$$\prod_{j=1}^{p}\left\{\frac{1}{\sqrt{2\pi}1000}e^{\frac{-\beta_j^2}{2*1000^2}}\right\} \tag{2.17}$$

Marginal for β

$$p(\beta|y,X) = \int_0^{\infty} p(\beta,\alpha|y,X)d\alpha \tag{2.18}$$

Marginal for α

$$p(\alpha|y,X) = \int_{-\infty}^{\infty} (\beta,\alpha|y,X)d\beta \tag{2.19}$$

where β is a vector of length $(p+1)$.

In the regression setting, closed form for the posterior distribution of β are generally not available, and therefore one needs to use numerical integration or Markov chain Monte Carlo methods. Before the advent of MCMC, numerical integration techniques were employed by Grieve (1987). However, due to the availability of computer software packages such as `LaplacesDemon`, the regression model in Equation 2.17, 2.18 and 2.19 can be fitted using different optimization methods (such as Trust region and Nelder-Mead) and various simulation algorithms (such as Random-walk Metropolis and independent Metropolis). These two methods can be used to solve the complex numerical integration including censoring mechanism by using `LaplaceApproximation` and `LaplacesDemon` functions. `LaplaceApproximation` is used for optimization and `LaplacesDemon` is used for simulation. In the next section, a breast cancer data has been described for the purpose of data analysis under the assumption GE distribution.

2.3 Survival data: prognosis of women with breast cancer

Breast cancer is one the most common form of cancer occurring in women living in Western world. The data given in Table 2.1 refers to the survival times (in months) of women who had received a simple or radical mastectomy to treat a tumour. The data is carried out at the Middlesex Hospital, and documented in Leathem and Brook (1987) and is also discussed by Collet (1994, 2003). In the table, the survival times of each woman are classified according to whether their tumour was positively or negatively stained. Censored survival times are labeled with an asterisk.

Negatively Stained: 23, 47, 69, 70*, 71*, 100*, 101*, 148, 181, 198*, 208*, 212*, 224*
Positively Stained: 5, 8, 10, 13, 18, 24, 26, 26, 31, 35, 40, 41, 48, 50, 59, 61, 68, 71, 76*, 105*, 107*, 109*, 113, 116*, 118, 143*, 154*, 162*, 188*, 212*, 217*, 225*

Table 2.1: *Survival times of women with tumours that were negatively or positively stained with HPA.*

2.4 Building Bayesian model of GE distribution with Laplace approximation

In this section, the main focus is on the specification of Bayesian model. A full theoretical description and summary of commands are provided, followed by details concerning calculations and graphics. Definition of creation of survival data, definition of Bayesian model which includes the prior and the likelihood specification are also provided. Bayesian fitting of GE model for this data can be done in `R` by using function `LaplaceApproximation` for analytic approximation and then with `LaplacesDemon` for MCMC simulations. Thus, implementation has been made by using `LaplacesDemon` package.

The full Bayesian model code in `R` to fit GE distribution is being described below.

2.4.1 Creation of breast cancer data

`LaplaceApproximation` function requires data that is specified in a list. Though most `R` functions use data in the form of a data frame, Laplace's Demon uses one or more numeric matrices in a list. It is much faster to process a numeric matrix than a data frame in iterative estimation. For the above data of **45** patients of prognosis of women with breast cancer has given the survival times of women with tumour that were negatively or positively stained with HPA.

```
library(LaplacesDemon)
y<-c(23,47,69,70,71,100,101,148,181,198,208,212,224,5,
   5,10,13,18,24,26,26,31,35,40,41,48,50,59,116,68,71,78,
   105,107,109,113,61,118,143,154,162,188,212,217,225)
x1<-c(rep(0,13),rep(1,32))
censor<-c(1,1,1,0,0,0,0,1,1,0,0,0,0,rep(1,18),0,0,0,0,1,
```

```
        0,1,1,rep(0,6))
X<-cbind(1,x1)
```

y is the vector of survival time containing both groups in it, x_1 is the indicator vector (0: negatively stained, 1: positively stained), censor is a binary vector of censoring using 1 for uncensored and 0 for censored observation. The matrix X is created by the function cbind termed as model matrix. Its first column is of $1's$ whereas second column x_1 is a column of indicator of staining. (0: negatively stained, 1: positively stained).

```
J<-2
mon.names<-c("LP","shape")
parm.names<-as.parm.names(list(beta=rep(0,J),log.shape=0))
MyData<-list(J=J,X=X,mon.names=mon.names,parm.names=parm.names,y=y,
censor=censor)
```

There are $J = 2$ independent variables (group1: negatively stained and group2: positively stained). The object mon.names is meant for the variables to be monitored. The R code defined above must have a name specified for each parameter in the vector parm.names, and parameter names must be included with the data in a list called as.parm.names. At the end of the above R code, the object called MyData has been created. MyData is the list of six vectors, namely, survival data vector y, censored survival observation vector censor, mon.names, parm.names, model matrix X and J. The user must specify the number of observations in the data as either a scalar n or N. If these are not found by the LaplaceApproximation or LaplacesDemon functions, then it will attempt to determine sample size as the number of rows in y or Y.

2.4.2 Generation of Initial values for generalized exponential distribution

The function LaplaceApproximation requires a vector of initial values for the parameters. Each initial value is a starting point for the estimation of a parameter. When all initial values are set to zero, LaplaceApproximation will optimize initial values using Nelder-Mead or trust region algorithm. Generally, parameter beta has been set equal to zero and parameter log.shape has been set equal to log(1), which is also zero. However, such guess value do not converge in most of the cases in this study. We have found that if regression cosfficient obtain from fitting simple

regression using log of the survival time as response often works better. Probably, the reason behind is the use of log link function for the scale parameter.

```
Initial.Values <-c(coef(lm(log(y)~x1)),log(1))
```

2.4.3 Model specification for GE distribution

The function `LaplaceApproximation` can fit any model in Bayesian aura for which likelihood and priors are specified. To use this function a model must be specified. Thus, for the fitting of the breast cancer data, consider that the survival time follows generalized exponential distribution, which is often written as,

$$y \sim GE(\alpha, \lambda)$$

Prior probabilities are specified for β and α.

$$\beta_j \sim N(0, 1000)$$

$$\alpha \sim HC(25)$$

Each component of the β parameter of length J is assigned a weak prior probability distribution that is normally distributed according to $\mu = 0$ and $\sigma = 1000$. The large variance or small precision indicates a lot of uncertainty about each β, and is hence a weak prior distribution. The shape parameter α is half-Cauchy-distributed according to its hyperparameter, scale$= 25$.

To specify a model, a function called `Model` must be created. The function `Model` contains the two main arguments, namely, `parm` and `Data`. The argument `parm` is the set of parameters and `Data` is the list of data. Then we start the specification of parameters i.e. beta and shape . Since `LaplaceApproximation` passes a vector of parameters called `parm` to `Model`, the function needs to know which parameter is associated with which element of `parm`. For this, the vector `beta` is declared, and then each element of `beta` is populated with the value associated in the corresponding element of `parm`. It is important to reparameterize all parameters to be real-valued. In the `Model` function each parameter must be unconstrained. Here, α receives a half-Cauchy distributed prior of the form:

$$\alpha \sim HC(25)$$

In this specification, α cannot be negative. By reparameterizing α as in line 4 of the code. After defining parameters the next step is to define priors for them. To work with the log of the prior densities and according to the assigned names of the

parameters and hyperparameters, they are specified in line **5** and **6**. Object `f1` and `s1` has been defined as the density and survival function of generalized exponential distribution, respectively. Finally, everything is put together to calculate `LP`, the logarithm of the unnormalized joint posterior density. The vector μ is the inner product of the design matrix, `Data$X`, and the transpose of the vector `beta`. The vector μ, vector `Data$y`, and scalar `shape` are used to estimate the sum of the log-likelihoods, where:

$$y \sim GE(\alpha, \lambda)$$

The function `Model` has been designed, is incredibly flexible, allowing a wide variety of Bayesian models to be specified. Hence, the full Bayesian model code for the regression analysis is described below:

```
Model<-function(parm,Data)
{
beta<-parm[1:Data$J]
shape<-exp(parm[Data$J+1])
beta.prior<-sum(dnorm(beta,0,1000,log=T))
shape.prior<-dhalfcauchy(shape,25,log=T)
mu<-tcrossprod(beta,Data$X)
scale<-exp(mu)
f1<-function(y,shape,scale) shape*dexp(y,1/scale)*
    pexp(y,1/scale)^(shape-1)
s1<-function(y,shape,scale) 1-pexp(y,1/scale)^shape
LL<-censor*log(f1(y,shape,scale))+
    (1-censor)*log(s1(y,shape,scale))
LL<-sum(LL)
LP<-LL+beta.prior+shape.prior
Modelout<-list(LP=LP,Dev=-2*LL,Monitor=c(LP,shape),
yhat=rgenexp(length(y),shape,scale),parm=parm)
return(Modelout)
}
```

2.4.4 Fitting of model with Laplace approximation

The Laplace approximation or Laplace method is a family of asymptotic techniques used to approximate integrals. Laplace's method accurately approximates unimodal posterior moments and marginal posterior distributions in many cases. `LaplaceApproximation` seeks a global maximum of the logarithm of the unnormalized joint posterior density. The approach differs by `Method`. In this section,

two optimization methods have been used to obtain the approximated posterior results using function `LaplaceApproximation`. The first method is trust region method Nocedal and Wright (1999) and second is Nelder and Mead (1965).

Trust-region methods define a region around the current iterate within which they trust the model to be an adequate representation of the objective function, and then choose the step to be the approximate minimizer of the model in this trust region. In effect, they choose the direction and length of the step simultaneously. If a step is not acceptable, they reduce the size of the region and find a new minimizer. In general, the step direction changes whenever the size of the trust region is altered.

The size of the trust region is critical to the effectiveness of each step. If the region is too small, the algorithm misses an opportunity to take a substantial step that will move it much closer to the minimizer of the objective function. If too large, the minimizer of the model may be far from the minimizer of the objective function in the region, so we may have to reduce the size of the region and try again. In practical algorithms, we choose the size of the region according to the performance of the algorithm during previous iterations. If the model is generally reliable, producing good steps and accurately predicting the behavior of the objective function along these steps, the size of the trust region is steadily increased to allow longer, more ambitious, steps to be taken. On the other hand, a failed step indicates that our model is an inadequate representation of the objective function over the current trust region, so we reduce the size of the region and try again. This method is implemented in `LaplaceApproximation` function with `"TR"` as a method. An object `M1` has been created as a result of using `LapalceApproximation` function. The posterior summaries obtained by trust region is reported in Table 2.2. The `R` code for the fitting of GE distribution is written as

```
M1<-LaplaceApproximation(Model,Initial.Values,Data=MyData,
    Iterations=10000,Method="TR")
```

The second method is Nelder-Mead algorithm. This method is used and implemented in the `LaplaceApproximation` function with object name `M2`. The posterior summaries obtained by Nelder-Mead algorithm is reported in Table 2.3.

```
M2<-LaplaceApproximation(Model,Initial.Values,Data=MyData,
    Iterations=10000,Method="NM")
```

Analytic- Trust region				
	Mode	SD	LB	UB
beta[1]	5.82	0.56	4.69	6.94
beta[2]	-0.96	0.51	-1.97	0.06
log.shape	-0.01	0.27	-0.54	0.52
Simulation- Sampling Importance Resampling				
	Mean	SD	LB	UB
beta[1]	6.02	0.57	5.03	7.22
beta[2]	-1.08	0.51	-2.18	-0.08
shape	0.95	0.22	0.60	1.47

Table 2.2: *The analytic and simualtion posterior summaries of breast cancer data under the assumption of generalized exponential disrtribution.*

Analytic- Nelder-Mead algorithm				
	Mode	SD	LB	UB
beta[1]	5.82	0.53	4.75	6.88
beta[2]	-0.95	0.51	-1.97	0.06
log.shape	-0.01	0.22	-0.46	0.43
Simulation- Sampling Importance Resampling				
	Mean	SD	LB	UB
beta[1]	5.99	0.57	5.01	7.28
beta[2]	-1.03	0.54	-2.19	-0.05
shape	0.95	0.23	0.59	1.47

Table 2.3: *The analytic and simualtion posterior summaries of breast cancer data under the assumption of generalized exponential disrtribution.*

2.4.5 Comparison of optimization techniques

In the previous subsection, breast cancer data has been analysed by two optimization techniques that are, trust region and Nelder-Mead optimization algorithms. It is to be found from Table 2.2 and 2.3 that they are very close in terms of numerical posterior summaries. But the algorithms differ in terms of convergence. Posterior density plots obtained by trust region and Nelder-Mead algorithms are reported in Figure 2.3, 2.4 and 2.5 and 2.6, respectively.

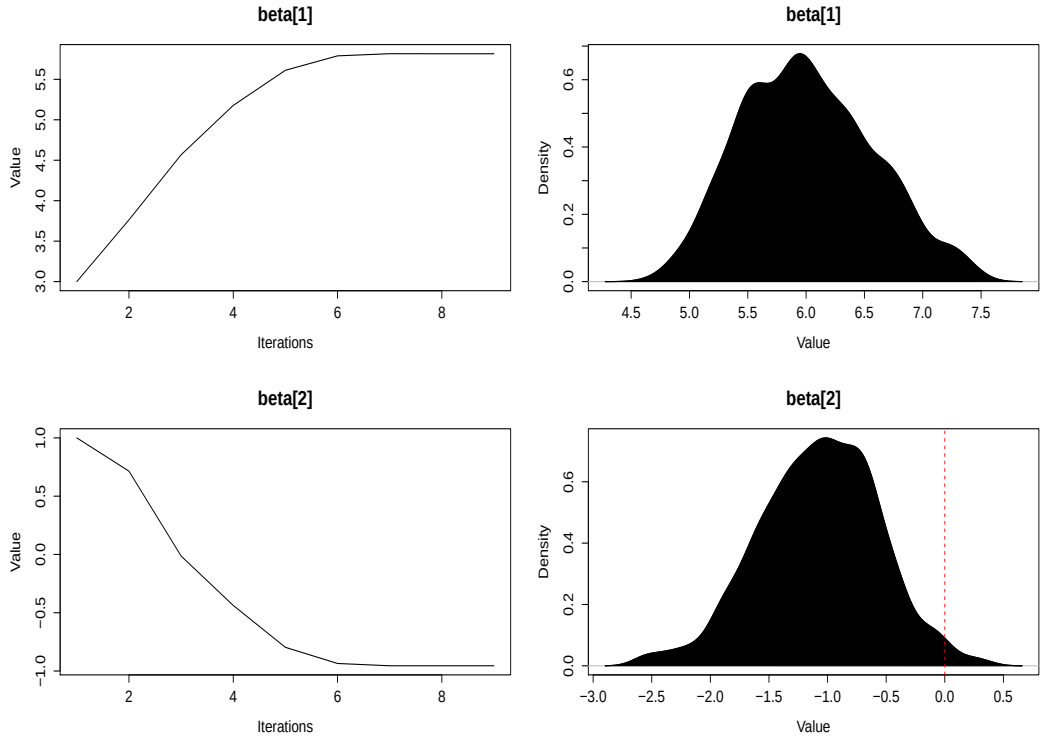

Figure 2.3: *Posterior density plots of regressor variables obtained by TR.*

In Figure 2.3, convergence of algorithm starts from 8th iteration. After this period the chain is stabilized within a zone. So the number of iterations we need to discard is 8 to monitor the sampled values which demonstrate much better behaviour with small periodicities. On the other hand, from Figure 2.5 the algorithm starts converges at 40th iteration. Here, the iterations to be discarded is around 40. The convergence speed of algorithm by N-M is much slower than from trust region algorithm. However, N-M is a simplex base optimization algorithm and does not require any derivatives, whereas, trust region method requires derivatives of the objective function. Notably, in the implementation of TR in `LaplaceApproximation`, supply of derivatives are not required.

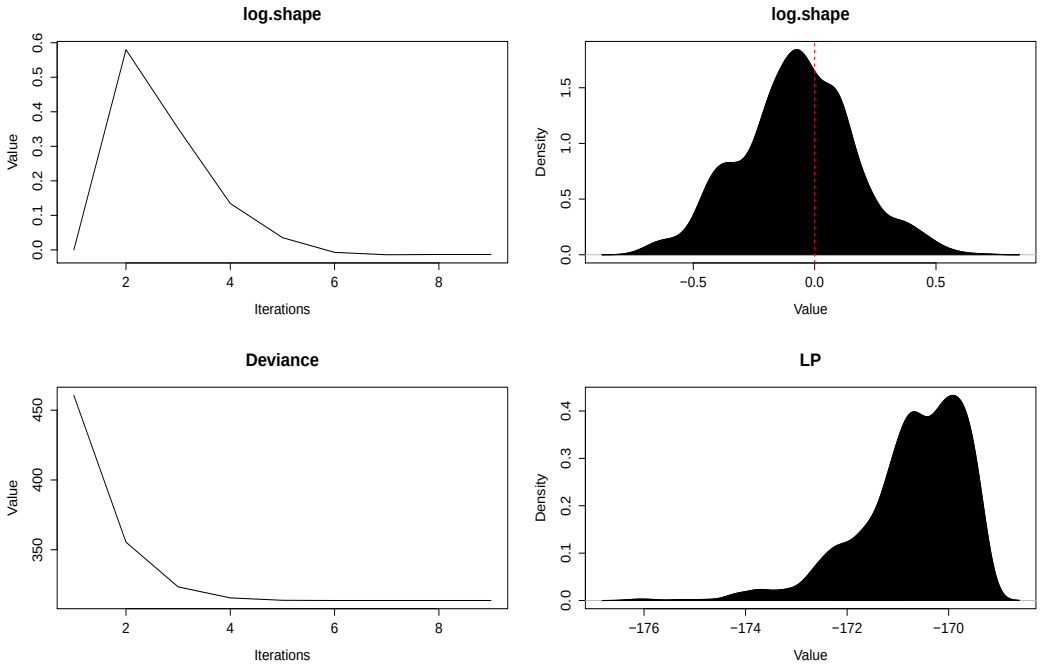

Figure 2.4: *Posterior density plots of regressor variables obtained by TR.*

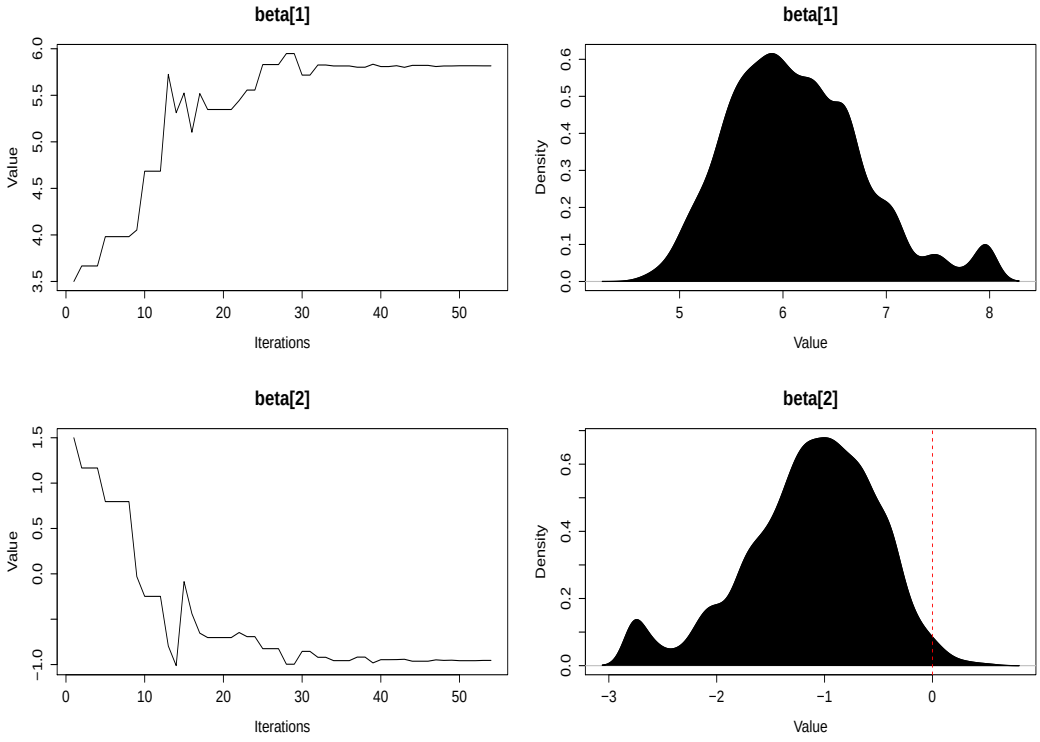

Figure 2.5: *Posterior density plots of regressor variables obtained by Nelder-Mead.*

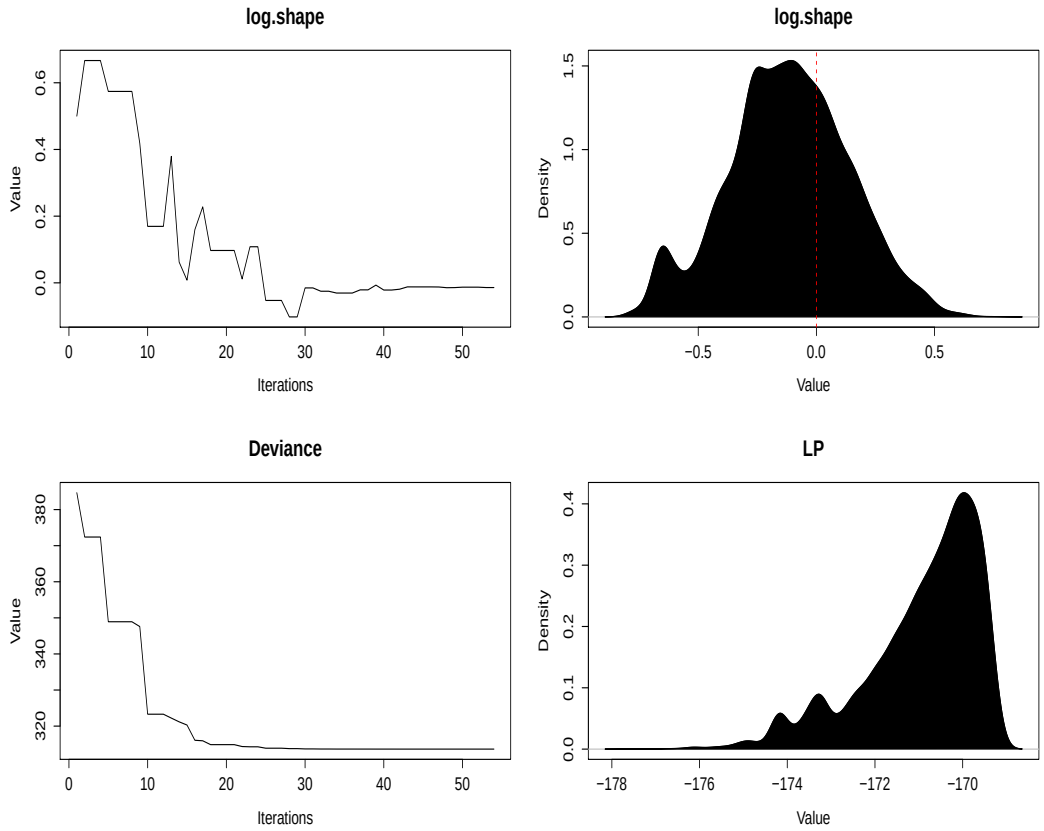

Figure 2.6: *Posterior density plots of regressor variables obtained by Nelder-Mead.*

2.4.6 Simulation study of breast cancer data under the assumption of GE model

In this section simulation will be performed by using two algorithms namely, random walk Metropolis algorithm and independent Metropolis algorithm. For the purpose of illustration breast cancer data has been used. The `R` commands for the implementation of RWM is given below with object name `M2` by using function `LaplacesDemon` and the results are summarized in Table 2.4 along with the histograms of generated values, and their corresponding kernel estimates of the posterior densities are depicted in Figure 2.7.

```
Initial.Values<-as.initial.values(M1)
M2<-LaplacesDemon(Model, Data=MyData, Initial.Values,
    Covar=M1$Covar, Iterations=50000, Status=100, Thinning=1,
    Algorithm="RWM",Specs=NULL)
M2
```

	Mean	SD	MCSE	ESS	LB	Median	UB
beta[1]	6.10	0.63	0.03	500.00	5.07	6.02	7.60
beta[2]	-1.10	0.58	0.03	435.24	-2.34	-1.09	-0.10
log.shape	-0.11	0.24	0.01	500.00	-0.58	-0.12	0.35
Deviance	316.66	2.44	0.10	500.00	313.95	316.04	323.10
LP	-170.77	1.22	0.05	500.00	-174.00	-170.45	-169.41
shape	0.92	0.22	0.01	385.71	0.56	0.89	1.42

Table 2.4: *Simulated posterior summaries of breast cancer data by Random-walk Metropolis algorithm under the assumption of GE model.*

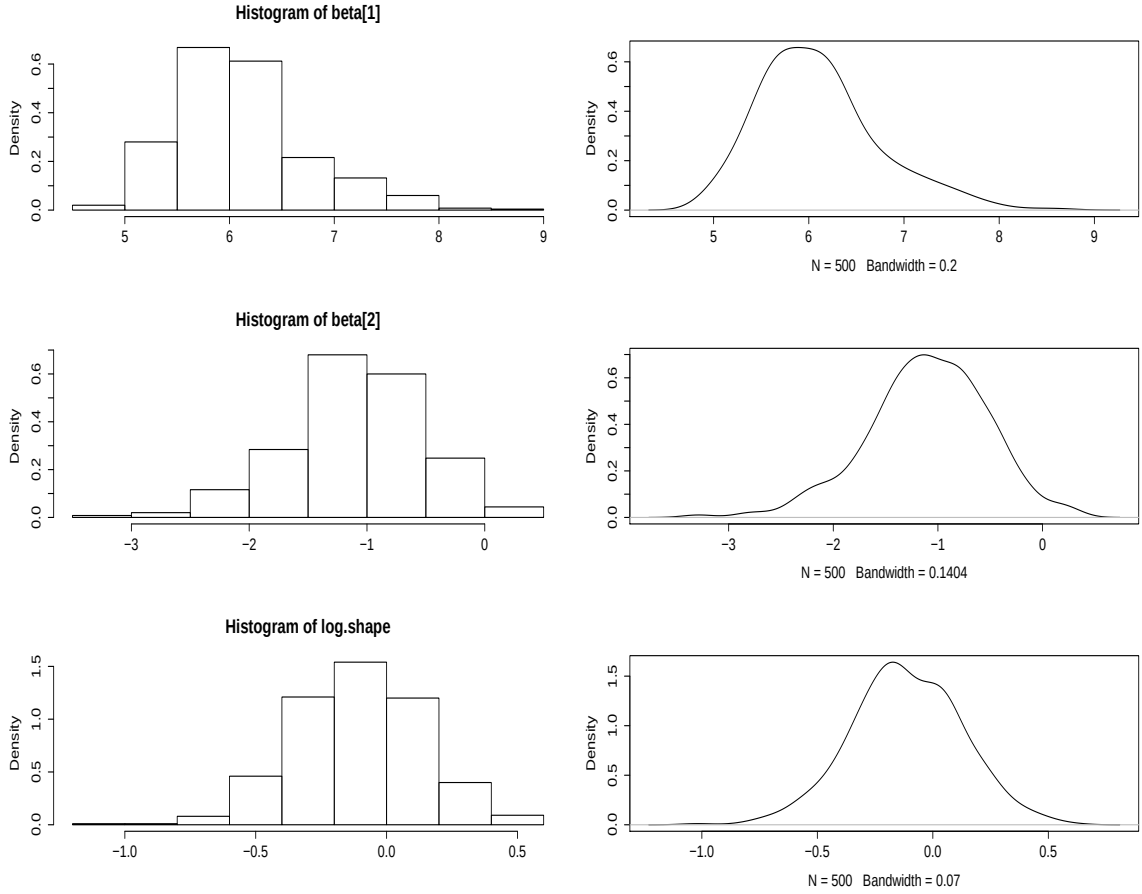

Figure 2.7: *Histogram and posterior density plots by random walk Metropolis.*

Now the second simulation algorithm used to get simulated posterior summary is independent Metropolis algorithm. The argument of `LaplacesDemon` function for the implementation of IM algorithm with object name `M3` is given as below:

```
Initial.Values<-as.initial.values(M1)
M3<-LaplacesDemon(Model, Data=MyData, Initial.Values,
Covar=M1$Covar, Iterations=20000, Status=100, Thinning=1,
Algorithm="IM",Specs=list(mu=M1$Summary1[1:length(Initial.Values),1]))
```

The output obtained by object `M3` is reported in Table 2.5 and its histograms and posterior density plots are reported in Figure 2.8.

	Mean	SD	MCSE	ESS	LB	Median	UB
beta[1]	5.85	0.33	0.01	2848.61	5.22	5.83	6.52
beta[2]	-0.96	0.30	0.01	2750.96	-1.57	-0.96	-0.39
log.shape	-0.03	0.15	0.00	3024.94	-0.33	-0.03	0.26
Deviance	314.73	0.92	0.03	2117.35	313.72	314.49	317.14
LP	-169.80	0.46	0.01	2115.89	-171.01	-169.68	-169.29
shape	0.98	0.15	0.00	2816.47	0.72	0.97	1.30

Table 2.5: *Simulated posterior summary obtained by independent Metropolis algorithm.*

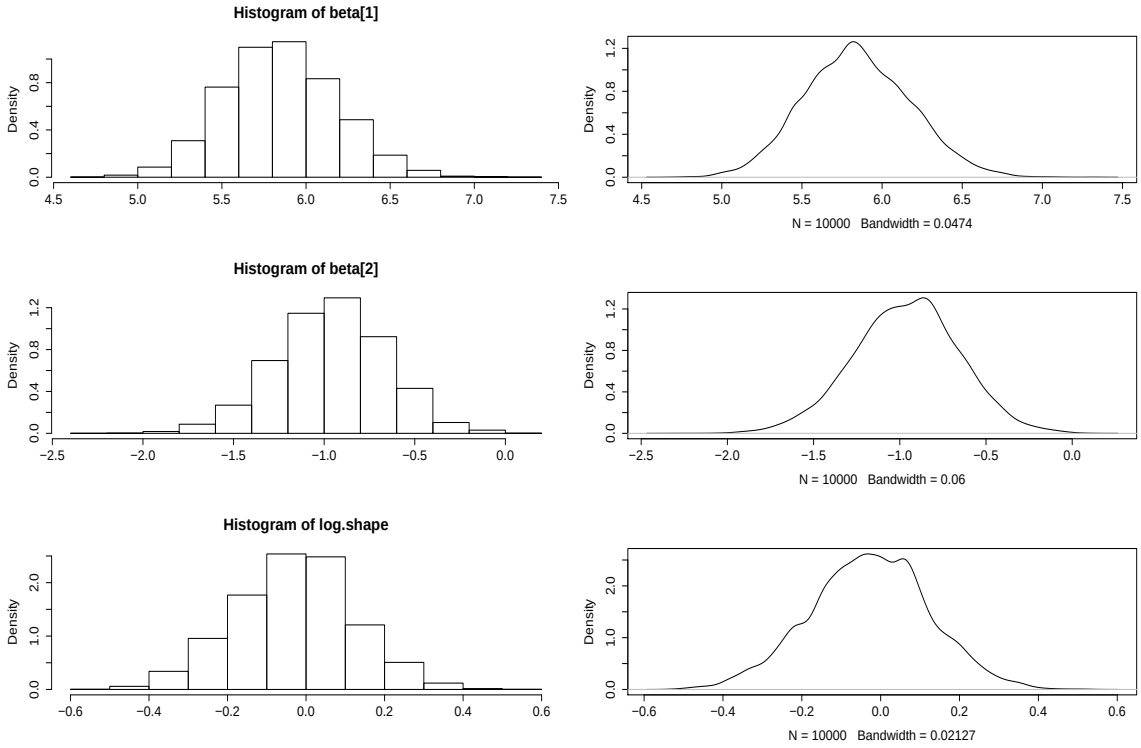

Figure 2.8: *Histograms and posterior density plots of parameter of generalized exponential distribution by independent Metropolis algorithm.*

2.4.7 Comparison of simulation techniques

The summary of the simulated posterior output obtained by using random walk Metropolis algorithm is reported in Table 2.4. This table consists of seven columns which contains posterior mean as well as posterior median. Third and fourth column of Table 2.4 is Monte Carlo standard error and effective sample size, respectively. Column fifth and seventh represent the **25%** quantile denoted by LB (Lower bound) and **97.5%** quantile denoted as UB (Upper bound), respectively. Here, it could seen that the value of MCSE is very small, which shows the convergence of algorithm. In random walk Metropolis algorithm the optimal acceptance rate according to Roberts et al. (1997) and Neal and Roberts (2008) is around **25%**, ranging from **0.23** for large dimensions to **0.45** for the univariate. Ntzoufras (2009) recommended tunning the variance of the proposal density such that the acceptance rate lies within the interval of **[20% – 40%]**, which are the values also proposed and used by Spiegelhalter et al. (2003). This range in accordance with the range

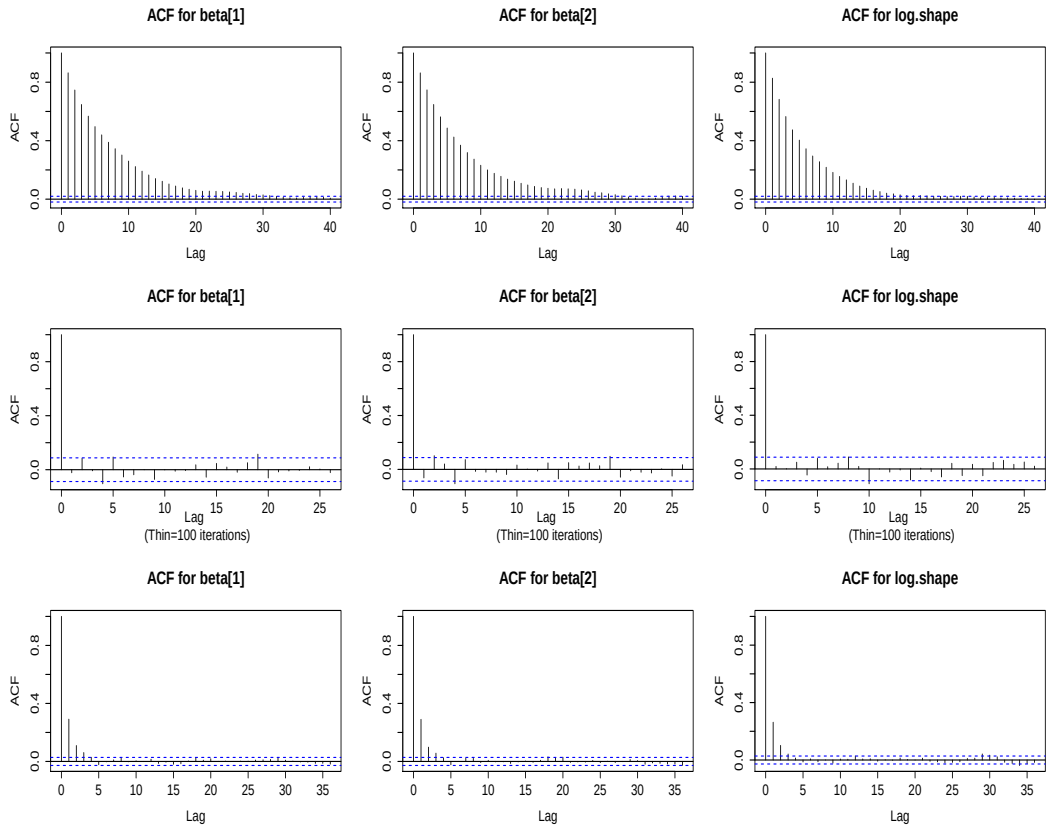

Figure 2.9: *Auto-correlations plots for generalized exponential regression parameters beta1 , beta2 and log.shape of breast cancer data using random-walk and independent Metropolis algorithm with independent normal proposals.*

of **[10% – 40%]** also suggested by Roberts and Rosenthal (2001). Here, in this breast cancer data the algorithm has been achieved an acceptance rate of **46%**

and for independent Metropolis algorithm it has **37.95%** acceptance probability. Random walk Metropolis algorithm was initially run for **10,000** iterations. The auto-correlation plots in the uppermost panel of Figure 2.9 indicate high correlation. For this reason, the number of iterations was increased to **50,000**. In order to eliminate high autocorrelations, a thinnig interval equal to 100 is considered. Then the low autocorrelation could be seen in the middlemost panel. The bottommost panel is autocorrelation of independent Metropolis algorithm, which already shows low correlation at **10000** iteration. Hence, the adequacy of these algorithms are confirmed by acceptance probability and autocorrelation plots. Thus, it would be concluded that random walk Metropolis algorithm and independent Metropolis algorithm both perform quite effectively for such a survival data under generalized exponential distribution.

2.4.8 Median survival time

Median survival time is the time beyond which **50%** of the individuals in the population under study are expected to survive, and is given by that value $t(50)$ which is such that $S(t(50)) = 0.5$. As we know survival time distribution is always positively skewed, the median is the preferred summary measure of the location of the distribution. Once the survivor function has been estimated, then it is easy to obtain an estimate of the median survival time.

The estimated median survival time, $\hat{t(50)}$, is defined to be the smallest observed survival time for which the value of the estimated survivor function is less than 0.5.

$$\hat{t(50)} = min\{t_i | S(\hat{t}_i) < 0.5\},$$

where t_i is the observed survival time for the ith individual, $i = 1, 2, \ldots, n$. Since the estimated survivor function only changes at a death time, this is equivalent to the definition

$$\hat{t(50)} = min\{t_{(j)} | S(\hat{t}_{(j)}) < 0.5\},$$

where $t_{(j)}$ is the jth ordered death time, $j = 1, 2, \ldots, r$.

2.4.8.1 Median survival time and other percentile of GE distribution

The median of a probability density function $f(t)$ is a point t_{med} on the real line which satisfies the equation

$$\int_{-\infty}^{t_{med}} f(t)dt = \frac{1}{2} \tag{2.20}$$

this implies that $F(t_{med}) = 1/2$. Hence for the GE distribution with survival function in Equation 2.3, $S(t_{med}; \alpha, \lambda) = 1 - F(t_{med}) = 1/2$ implies

$$S[t_{med}] = 1 - \left(1 - exp\left(-\frac{t_{med}}{\lambda}\right)\right)^{\alpha} = 0.5$$

This gives,

$$\Longrightarrow t_{med} = -\lambda \log[1 - (0.5)^{1/\alpha}]$$

Consider the GE distribution, the $100p$-percentage point is obtained by equating the cumulative probability distribution function to p, where $0 \leq p \leq 1$.
That is,

$$F(t_p) = p$$

$$\Longrightarrow \left(1 - exp\left(-\frac{t_p}{\lambda}\right)\right)^{\alpha} = p.$$

Solving for t_p gives

$$t_p = -\lambda \log\left[1 - \left(\frac{100 - p}{100}\right)^{1/\alpha}\right]. \tag{2.21}$$

This gives the value of the point t_p on the real line that produce a percentage p of the distribution. We can easily test this by checking the value of t_p when $p = 50$ which corresponds to the median.

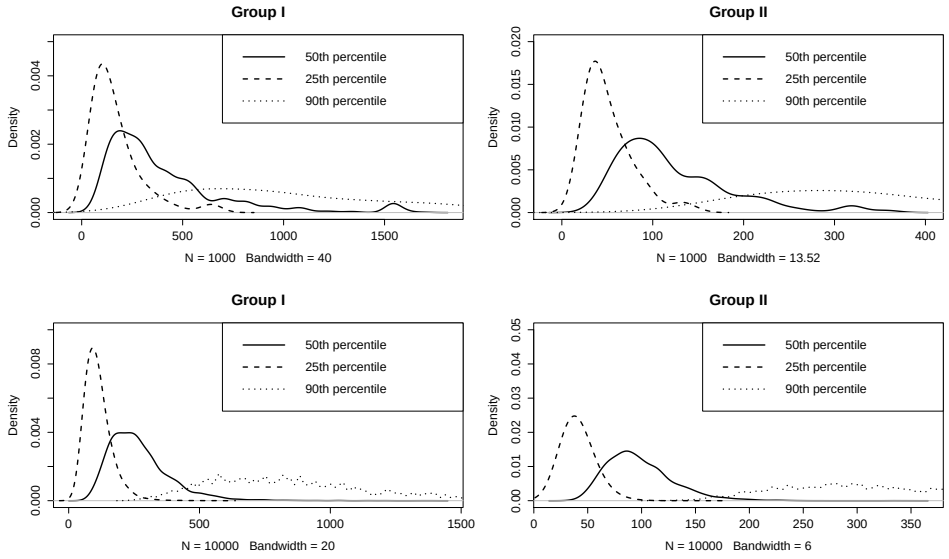

Figure 2.10: *Simulated posterior density plots of median and other percentile for both groups. The upper panel of the figure is obtained by SIR and lower panel is from independent Metropolis algorithm.*

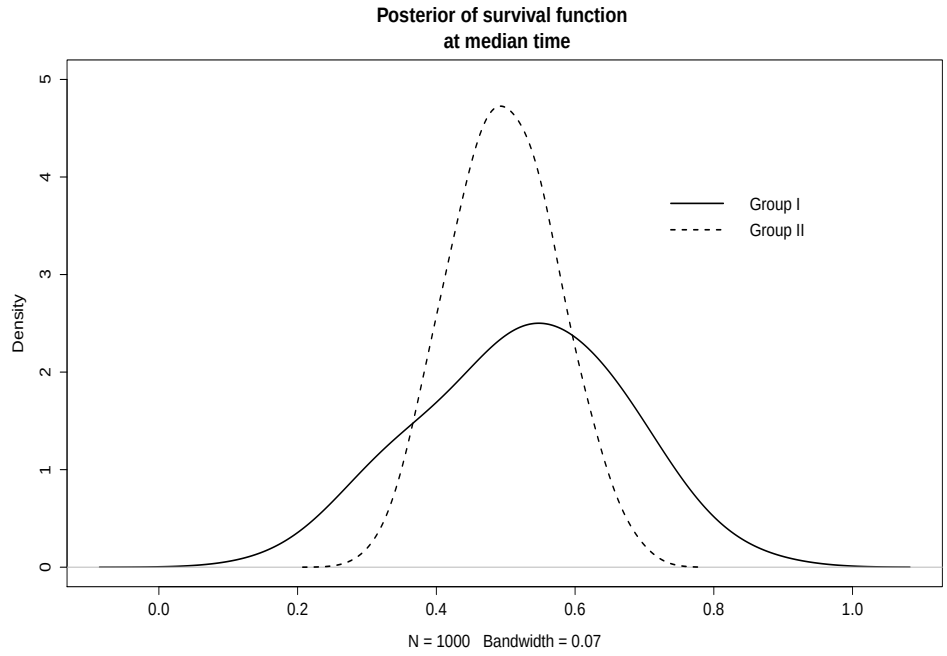

Figure 2.11: *Posterior density plots of survival function at* $t_{med} = 247$ *for group 1 and* $t_{med} = 95$ *for group 2.*

	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
Negatively stained	0.12	0.42	0.53	0.52	0.62	0.87
Positively stained	0.30	0.44	0.50	0.50	0.55	0.69

Table 2.6: *Survival probabilities of breast cancer data at median time for both groups for generalized exponential distribution.*

The frequentist counter part of this Bayesain analysis for $S(t)$ is based on some adhock methods and it is sometimes seems difficult to choose a better method Tableman and Kim (2004, p. 32) and Collet (1994, p. 33 and 34). Contrary to this, Bayesian approach quite straight forward and does not require any asymptotic approximation. Here, for breast cancer data, Table 2.6 and Figure 2.11 clearly shows that survival probability of negatively stained tumour is 0.53 and for positively stained it is 0.50. Meaning, survival probability of women having negatively stained tumour is more than the women having positively stained tumour.

The Kaplan-Meier estimate of the survivor function, for each of two groups of survival times, is plotted in Figure 2.12, which are in the form of step plots. The plots of fitted survival curves are also depicted in the same figure by smooth curve. Notice that in this figure, the Kaplan-Meier estimates extend to the time of the

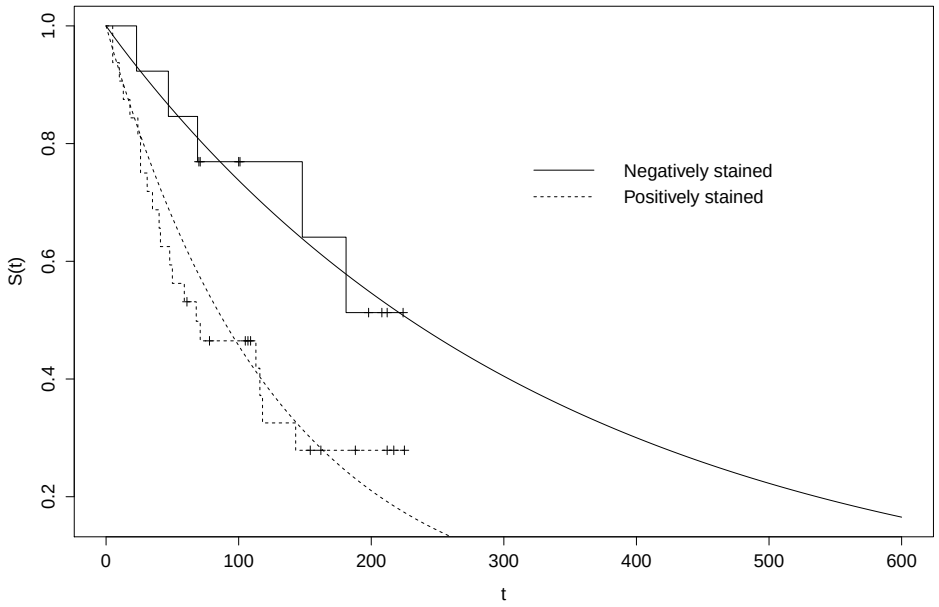

Figure 2.12: *Comparison of Kaplan-Meier estimate of survivor functions with generalized exponential survivor function for women with tumour that were positively and negatively stained. The closeness of the survival curves obtained by the non-parametric and Bayesian method is self explanatory.*

largest censored observation in each group. This figure shows that the estimated survivor function for those women with negatively stained tumours is always greater than that for women with positively stained tumours. This means that at any time t, the estimated probability of survival beyond t is greater for women with negatively staining, suggesting that the result of the HPA staining procedure might be a useful prognostic indicator. In particular, those women whose tumours are positively stained appear to have a poorer prognosis than those with negatively stained tumours. The high survival probabilities for negatively stained than the positively stained groups is quite evident from these survival curves. The degree of closeness between the fitted model and the non-parametric Kaplan-Meier method is also evident from this figure.

2.5 Fitting of breast cancer data with exponential extension distribution

Another form of exponential distribution which would be analyse in Bayesian framework is exponential extention distribution. This distribution is already discussed in Section 2.1. The likelihood function for exponential distribution from Equation 2.5

and 2.6,

$$L = \prod_{i=1}^{n} \left[\left(\frac{\alpha}{\lambda} \left(1 + \frac{y_i}{\lambda}\right)^{\alpha-1} .exp\left\{x_i^T \beta\right\} \right)^{\delta_i} \left(exp\left(1 - \left(1 + \frac{t_{c_i}}{\lambda}\right)^{\alpha}\right) exp\left\{x_i^T \beta\right\} .t_{c_i} \right)^{1-\delta_i} \right]$$

Prior,

$$\alpha \sim \text{half-Cauchy}(\gamma)$$

$$\beta_j \sim N(0, 1000)$$

Then, the joint posterior distribution would be,

$$p(\beta, \alpha | y, X) = \prod_{i=1}^{n} \left[\left(\frac{\alpha}{\lambda} \left(1 + \frac{y_i}{\lambda}\right)^{\alpha-1} .exp\left\{x_i^T \beta\right\} \right)^{\delta_i} \left(exp\left(1 - \left(1 + \frac{t_{c_i}}{\lambda}\right)^{\alpha}\right) exp\left\{x_i^T \beta\right\} .t_{c_i} \right)^{1-\delta_i} \right]$$

$$\left\{ \frac{2\gamma}{\pi(\alpha^2 + \gamma^2)} \right\} . \prod_{j=1}^{p} \left\{ \frac{1}{\sqrt{2\pi}1000} e^{\frac{-\beta_j^2}{2*1000^2}} \right\} \tag{2.22}$$

The implementation of analytic approximation and simulation is done via `LaplacesDemon` package. Here again trust region and Nelder-Mead optimization algorithms have been used to get approximated results. Table 2.7 represents the posterior summaries obtained by trust region method and Table 2.8 is the output obtained by Nelder-Mead. Simulated posterior output by random walk Metropolis and independent Metropolis algorithms are summarized in Table 2.9.

```
y<-c(23,47,69,70,71,100,101,148,181,198,208,212,224,
   5,5,10,13,18,24,26,26,31,35,40,41,48,50,59,116,68,
   71,78,105,107,109,113,61,118,143,154,162,188,212,217,225)
x1<-c(rep(0,13),rep(1,32))
censor<-c(1,1,1,0,0,0,0,1,1,0,0,0,0,rep(1,18),0,0,0,0,
        1,0,1,1,rep(0,6))
X<-cbind(1,x1)
J<-2
mon.names<-c("LP","shape")
parm.names<-as.parm.names(list(beta=rep(0,J),log.shape=0))
MyData<-list(J=J,X=X,mon.names=mon.names,parm.names=parm.names,
        y=y,censor=censor)
Initial.Values <-c(coef(lm(log(y)~x1)),log(1))
Model<-function(parm,Data)
{
beta<-parm[1:Data$J]
```

```
shape<-exp(parm[Data$J+1])
beta.prior<-sum(dnorm(beta,0,1000,log=T))
shape.prior<-dhalfcauchy(shape,25,log=T)
mu<-tcrossprod(beta,Data$X)
scale<-exp(mu)
f1<-function(y,shape,scale) shape/scale*(1+y/scale)^(shape-1)*
    exp(1-(1+y/scale)^shape)
s1<-function(y,shape,scale) exp(1-(1+y/scale)^shape)
LL<-censor*log(f1(y,shape,scale))+(1-censor)*
    log(s1(y,shape,scale))
LL<-sum(LL)
LP<-LL+beta.prior+shape.prior
Modelout<-list(LP=LP,Dev=-2*LL,Monitor=c(LP,shape),
          yhat=rexpext(length(y),shape,scale),parm=parm)
return(Modelout)
}

M1<-LaplaceApproximation(Model,Initial.Values,Data=MyData,
    Iterations=10000,Method="TR")
M1
```

Analytic- Trust region				
	Mode	SD	LB	UB
beta[1]	5.00	0.80	3.40	6.59
beta[2]	-1.14	0.61	-2.36	0.08
log.shape	-0.65	0.46	-1.57	0.28
Simulation-Sampling Importance Resampling				
	Mean	SD	LB	UB
beta[1]	5.12	0.92	3.25	6.90
beta[2]	-1.20	0.65	-2.65	-0.04
shape	0.65	0.42	0.23	1.89

Table 2.7: *The analytic and simualtion posterior summaries of breast cancer data under the assumption of exponential extension distribution.*

```
M2<-LaplaceApproximation(Model,Initial.Values,Data=MyData,
    Iterations=10000,Method="NM")
```

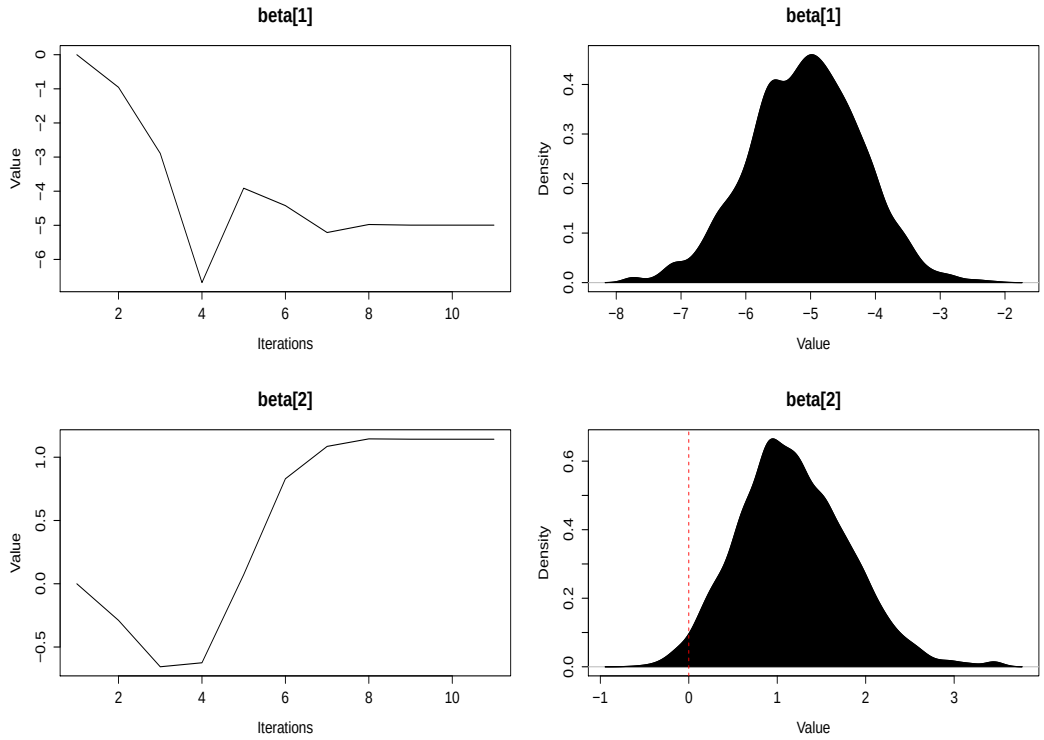

Figure 2.13: *Posterior density plots of regressor variables obtained by TR.*

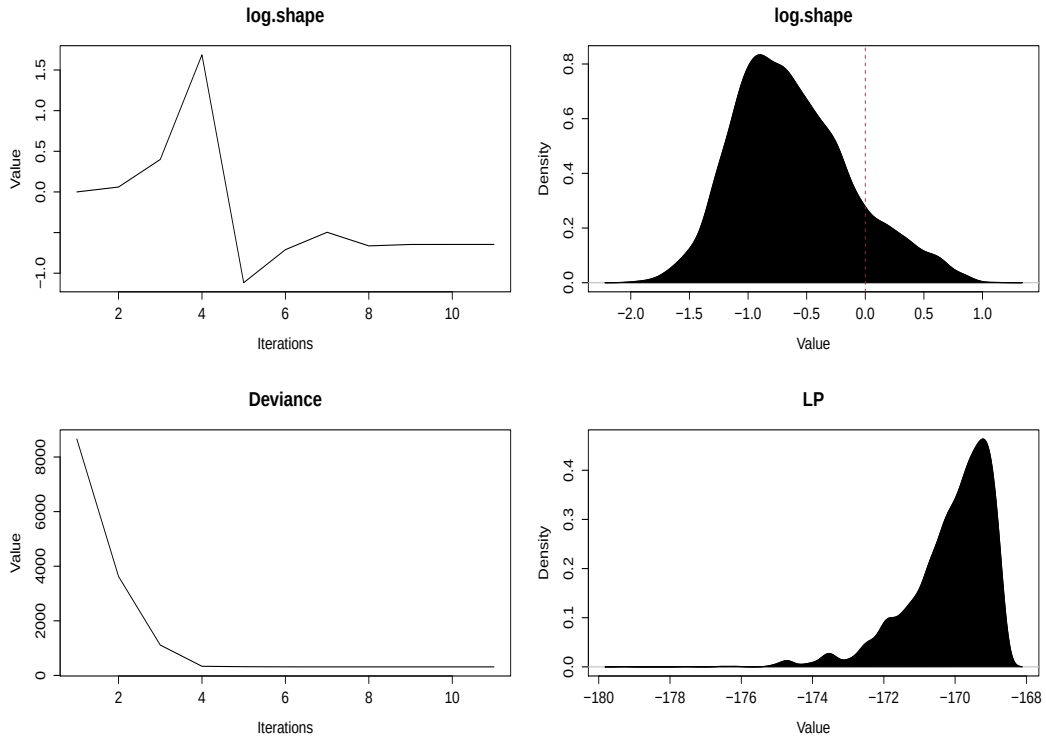

Figure 2.14: *Posterior density plots of regressor variables obtained by TR.*

M2

Analytic- Nelder-Mead algorithm				
	Mode	SD	LB	UB
beta[1]	5.00	0.80	3.40	6.59
beta[2]	-1.14	0.61	-2.36	0.08
log.shape	-0.64	0.46	-1.57	0.28
Simulation- Sampling Importance Resampling				
	Mean	SD	LB	UB
beta[1]	5.17	0.94	3.46	7.13
beta[2]	-1.19	0.61	-2.55	-0.13
shape	0.72	0.56	0.24	2.75

Table 2.8: *The analytic and simualtion posterior summaries of breast cancer data under the assumption of exponential extension disrtribution.*

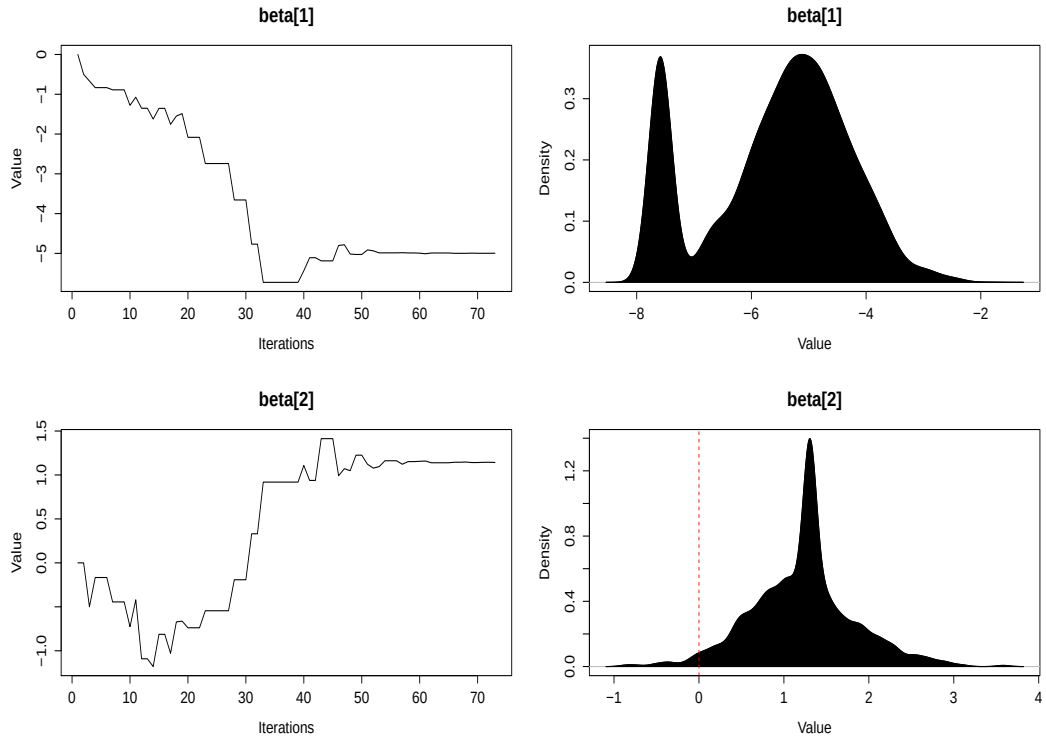

Figure 2.15: *Posterior density plots of regressor variables obtained by NM.*

Figure 2.13, 2.14 and 2.15, 2.16 are the trace and density plots of EE distribution obtained by trust region method and Nelder-Mead, respectively. For the fitting of EE distribution, trust region method has better performance as compared to N-M method. Trust region algorithm has converged fast at 8th iteration whereas

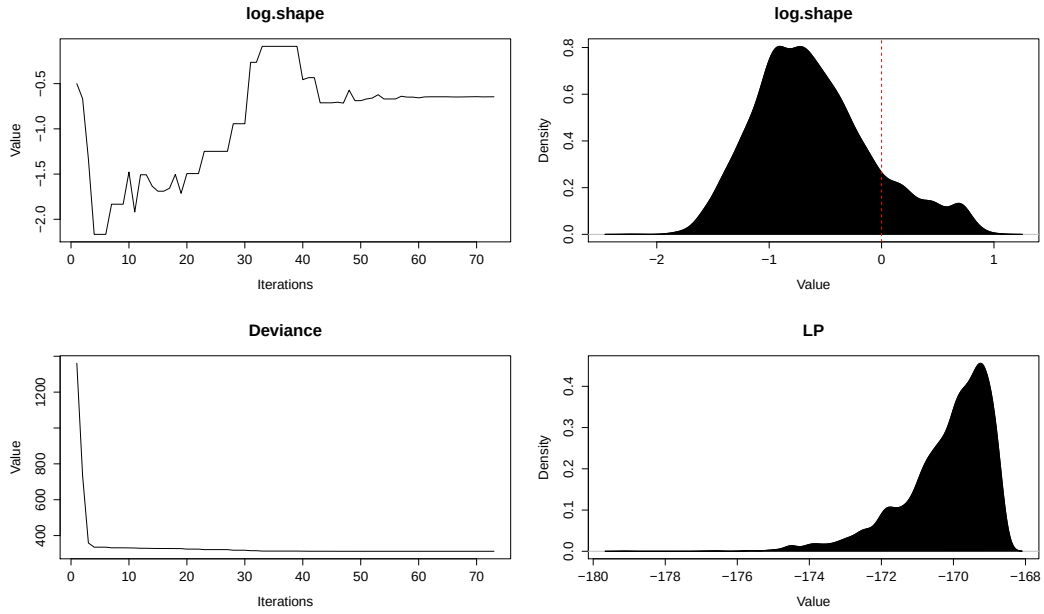

Figure 2.16: *Posterior density plots of regressor variables obtained by NM.*

N-M has converged at around 70th iteration. Again trust region method shows its superiority in terms of fitting such survival data.

2.5.1 Simulation study for exponential extension distribution

This section includes the Bayesian fitting of exponential extension distribution by random walk and independent Metropolis algorithm. `M2` is the object assigned for the implementation of random walk Metropolis algorithm and `M3` is the object for independent Metropolis algorithm. The numerical simulated posterior summaries of both algorithms are reported in Table 2.9. Here, random walk Metropolis algorithm was initially run for 20,000 iterations, which indicates high correlation as evident from uppermost panel of Figure 2.17. So, in order to make fast convergence of algorithm the iterations was increased to 50,000 and a thinning interval is taken as 100 as given in object `M2`. Then, the low correlation plot could be seen in the middlemost panel of Figure 2.17. The bottommost panel of Figure 2.17 is the auto-correlation plot for independent Metropolis algorithm which shows fast convergence as it gives low auto-correlation plots.

```
Initial.Values<-as.initial.values(M1)
M2<-LaplacesDemon(Model, Data=MyData, Initial.Values,
    Covar=M1$Covar, Iterations=50000, Status=F, Thinning=100,
```

```
    Algorithm="RWM",Specs=NULL)
M2

Initial.Values<-as.initial.values(M1)
M3<-LaplacesDemon(Model, Data=MyData, Initial.Values,
Covar=M1$Covar, Iterations=20000, Status=F, Thinning=3,
Algorithm="IM",Specs=list(mu=M1$Summary1[1:length(Initial.Values),1]))
M3
```

Random-walk Metropolis algorithm							
	Mean	SD	MCSE	ESS	LB	Median	UB
beta[1]	5.27	1.10	0.06	375.53	3.39	5.15	7.73
beta[2]	-1.22	0.66	0.03	336.68	-2.68	-1.18	-0.15
shape	1.05	3.34	0.18	387.93	0.24	0.53	5.26
Independent-Metropolis algorithm							
beta[1]	5.01	0.48	0.01	4922.15	4.07	5.01	5.94
beta[2]	-1.15	0.36	0.01	4912.67	-1.86	-1.15	-0.45
shape	0.55	0.16	0.00	4738.06	0.31	0.52	0.92

Table 2.9: *Simulated posterior summaries obtained by random walk and independent Metropolis algorithm under the assumption of exponential extension distribution.*

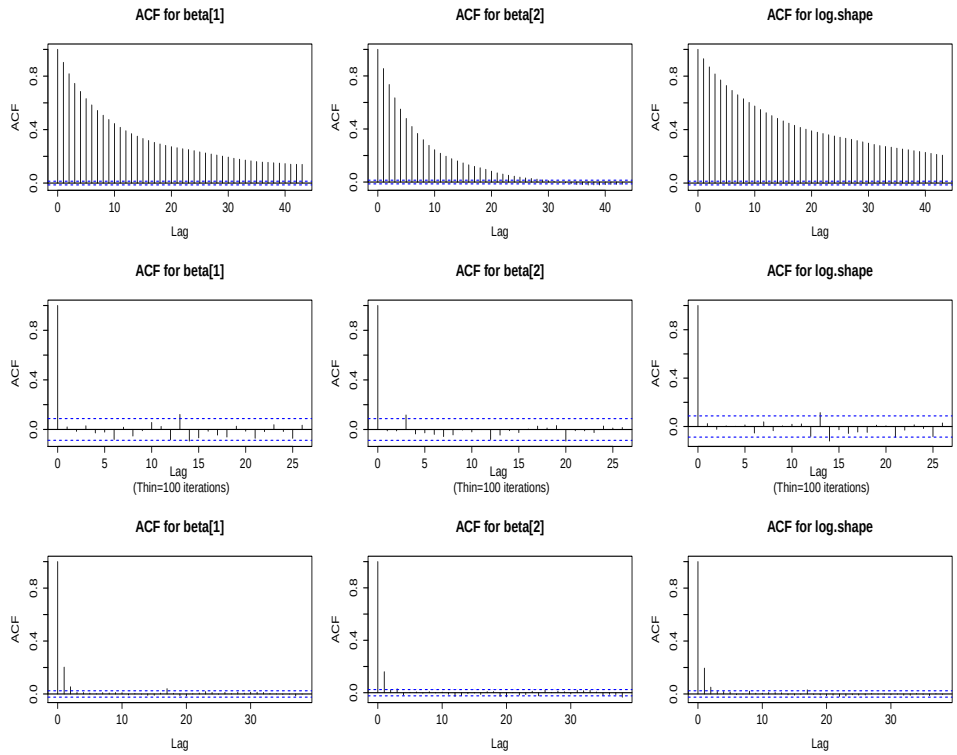

Figure 2.17: *Auto-correlations plots for exponential extension regression parameters beta1 , beta2 and log.shape of breast cancer data using random-walk and independent Metropolis algorithm with independent normal proposals.*

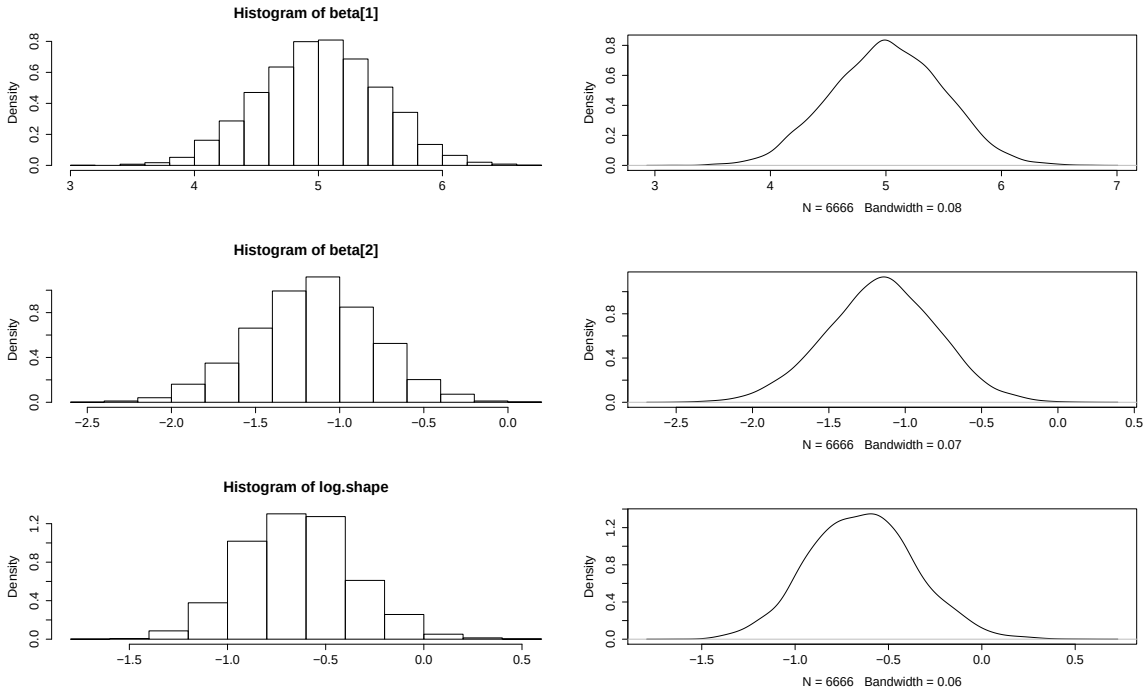

Figure 2.18: *Histograms and density plots by RWM.*

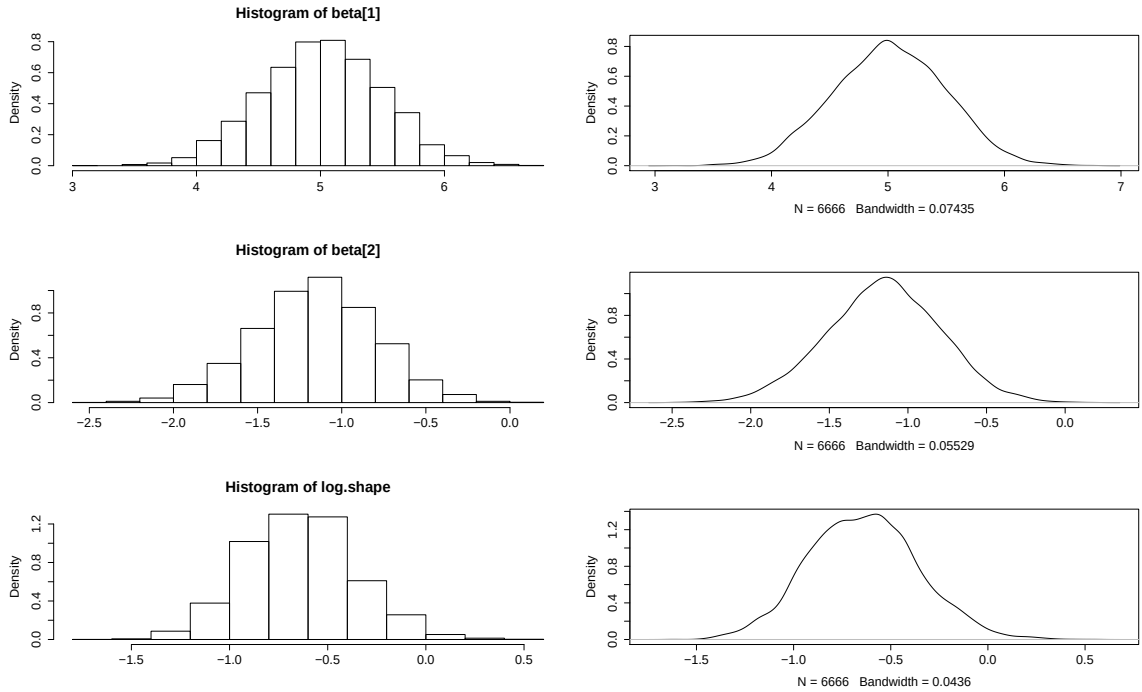

Figure 2.19: *Histograms and density plots by IM.*

2.5.2 Median and percentiles of exponential extension distribution

$$S(t_{med}) = exp\left(1 - \left(1 + \frac{t}{\lambda}\right)^{\alpha}\right) = 0.5$$

This gives,

$$t_{med} = \lambda[(1 - \log(0.5))^{1/\alpha} - 1]$$

the $100p$-percentage point is obtained by equating the cumulative probability distribution function to p, where $0 \leq p \leq 1$.
That is,

$$F(t_p) = p$$

$$\Longrightarrow exp\left(1 - \left(1 + \frac{t_p}{\lambda}\right)^{\alpha}\right) = p.$$

Solving for t_p gives

$$t_p = \lambda \left[\left(1 - \log\left(\frac{100 - p}{100}\right)\right)^{1/\alpha} - 1\right] \tag{2.23}$$

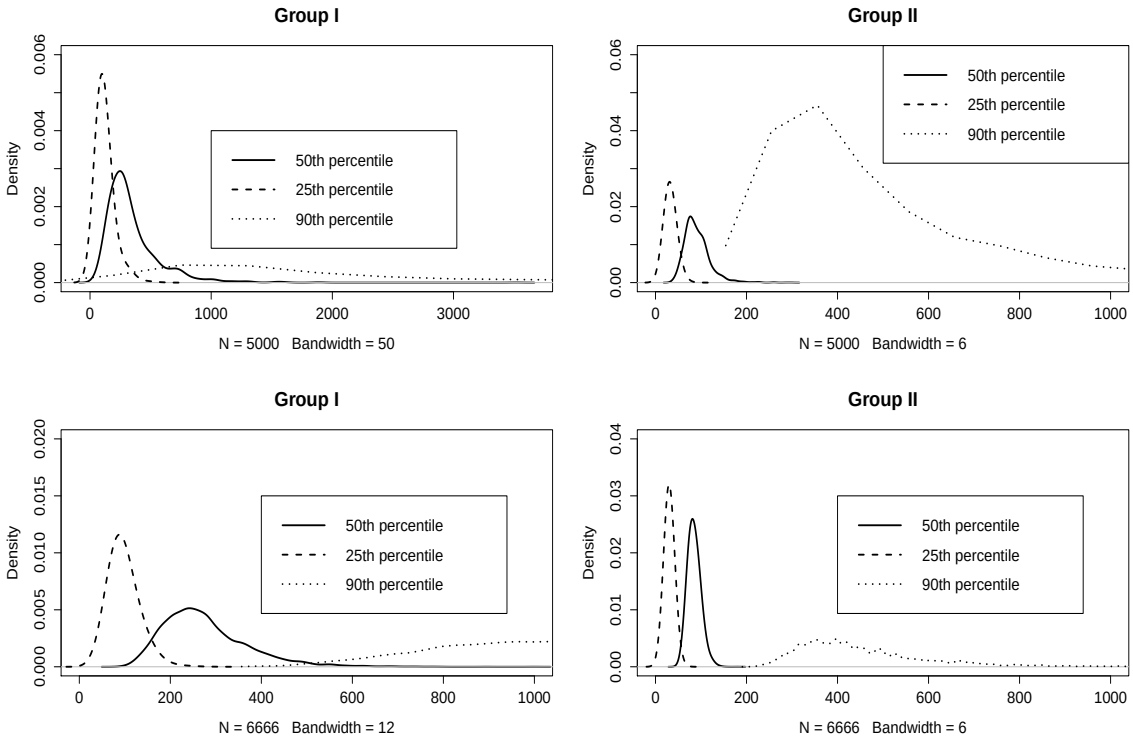

Figure 2.20: *Median and other percentile of exponential extension distribution.*

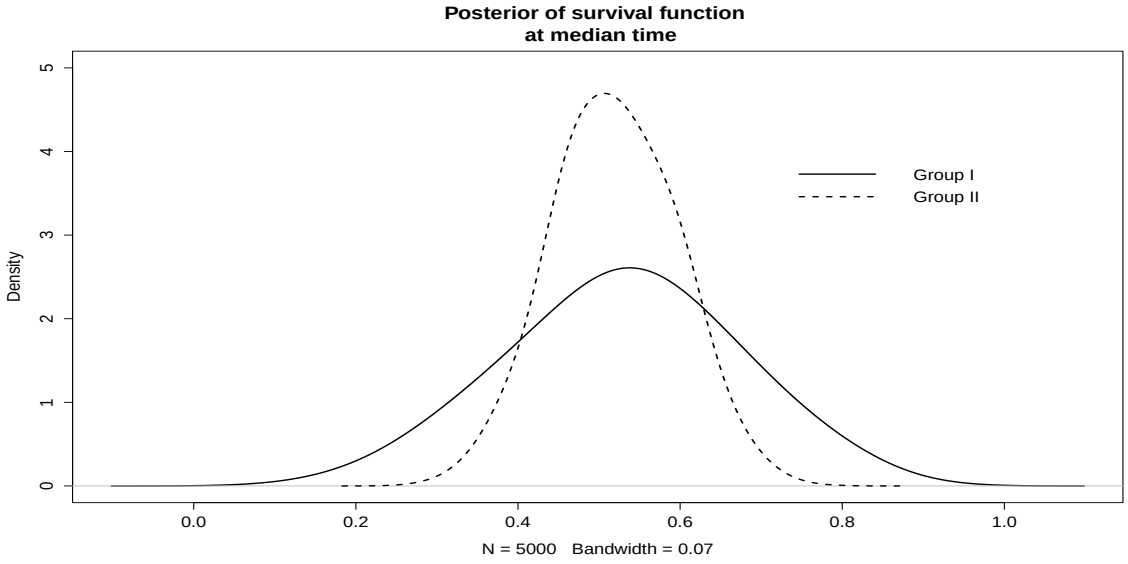

Figure 2.21: *Posterior density plots of survival function at* $t_{med} = 255$ *for group 1 and* $t_{med} = 81$ *for group 2.*

In the next section, Bayesian modelling of exponential distribution will be performed following the same pattern which has done in the previous sections. No more theoretical description has been made just to avoid the repetition.

	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
Negatively stained	0.11	0.44	0.54	0.53	0.62	0.89
Positively stained	0.27	0.46	0.52	0.52	0.58	0.78

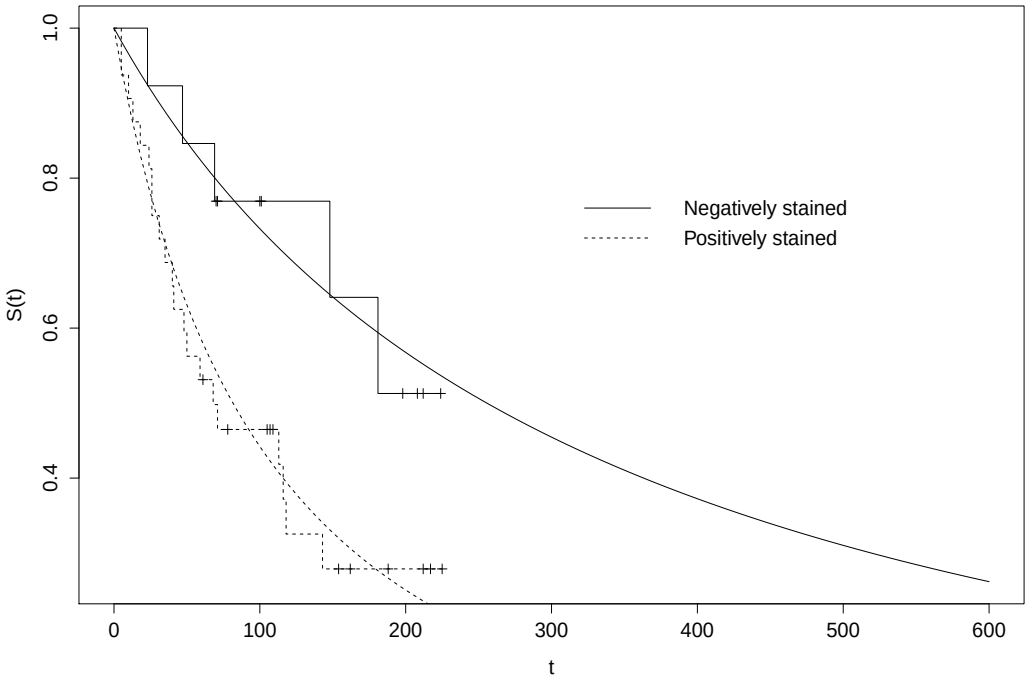

Figure 2.22: *Comparison of Kaplan-Meier survival curves with exponential extension survival curves.*

2.6 Fitting of exponential distribution, Special case: when $\alpha = 1$

When shape is taken as 1 in both GE & EE distribution, it will result the exponential distribution with one parameter.The likelihood function of exponential distribution would be,

$$L = \prod_{i=1}^{n} \left[\left(\frac{1}{\lambda} . (exp\left(-\frac{y_i}{\lambda}\right) exp(x_i^T \beta) \right)^{\delta_i} \left(exp\left(-\frac{t_{c_i}}{\lambda}\right) . exp(x_i^T \beta) . t_{c_i} \right)^{1-\delta_i} \right]$$

Prior,

$$\beta_j \sim N(0, 1000)$$

Then the joint posterior distribution would be,

$$p(\beta,\lambda|y,X)=\prod_{i=1}^{n}\left[\left(\frac{1}{\lambda}.(exp\left(-\frac{y_i}{\lambda}\right)exp(x_i^T\beta)\right)^{\delta_i}\left(exp\left(-\frac{t_{c_i}}{\lambda}\right).exp(x_i^T\beta).t_{c_i}\right)^{1-\delta_i}\right].$$

$$\prod_{j=1}^{p}\left\{\frac{1}{\sqrt{2\pi}1000}e^{\frac{-\beta_j^2}{2*1000^2}}\right\} \tag{2.24}$$

Following are the R codes for the implementation of exponential distribution.

```
y<-c(23,47,69,70,71,100,101,148,181,198,208,212,224,5,5,10,
   13,18,24,26,26,31,35,40,41,48,50,59,116,68,71,78,105,107,
   109,113,61,118,143,154,162,188,212,217,225)
censor<-c(1,1,1,0,0,0,0,1,1,0,0,0,0,rep(1,18),0,0,0,0,
        1,0,1,1,rep(0,6))
x1<-c(rep(0,13),rep(1,32))
X<-cbind(1,x1)
J<-2
mon.names<-c("LP")
parm.names<-as.parm.names(list(beta=rep(0,J)))
MyData<-list(J=J,X=X,mon.names=mon.names,parm.names=parm.names,
        y=y,censor=censor)
Initial.Values<-c(rep(0,J))
Model<-function(parm,Data)
{
beta<-parm[1:Data$J]
beta.prior<-sum(dnorm(beta,0,sqrt(1000),log=T))
mu<-tcrossprod(beta,Data$X)
theta<-exp(mu)
LL<-sum(censor*dexp(Data$y,rate=1/theta,log=T)+
    (1-censor)*pexp(Data$y,rate=1/theta,log.p=T,
    lower.tail=FALSE))
LP<-LL+beta.prior
Modelout<-list(LP=LP,Dev=-2*LL,Monitor=c(LP),yhat=
          rexp(length(y,theta),parm=parm))
return(Modelout)
}
M1<-LaplaceApproximation(Model,Initial.Values,Data=MyData,
    Iterations=10000,Method="TR")
```

The output obtained by object `M1` which is the implementation of trust region optimization method is reported in Table 2.10. Object `M2` is the implementation of Nelder-Mead method whose output is reported in Table 2.11. Simulated posterior summaries obtained by random walk and independent Metropolis algorithm are reported in Table 2.12.

Analytic-Trust region				
	Mode	SD	LB	UB
beta[1]	5.79	0.45	4.91	6.69
beta[2]	-0.95	0.50	-1.94	0.04
Simulation-Sampling Importance Resampling				
	Mean	SD	LB	UB
beta[1]	5.88	0.44	-6.83	-5.07
beta[2]	-1.01	0.54	-2.08	-0.01

Table 2.10: *The analytic and simualtion posterior summary of breast cancer data under the assumption of exponential disrtribution.*

```
M2<-LaplaceApproximation(Model,Initial.Values,Data=MyData,Iterations=10000,
Method="NM")
```

Analytic- Nelder-Mead algorithm				
	Mode	SD	LB	UB
beta[1]	5.80	0.45	4.91	6.69
beta[2]	-0.95	0.50	-1.95	0.04
Simulation- Sampling Importance Resampling				
	Mean	SD	LB	UB
beta[1]	5.87	0.47	5.08	7.03
beta[2]	-1.00	0.52	-2.21	-0.09

Table 2.11: *The analytic and simualtion posterior summary of breast cancer data under the assumption of exponential disrtribution.*

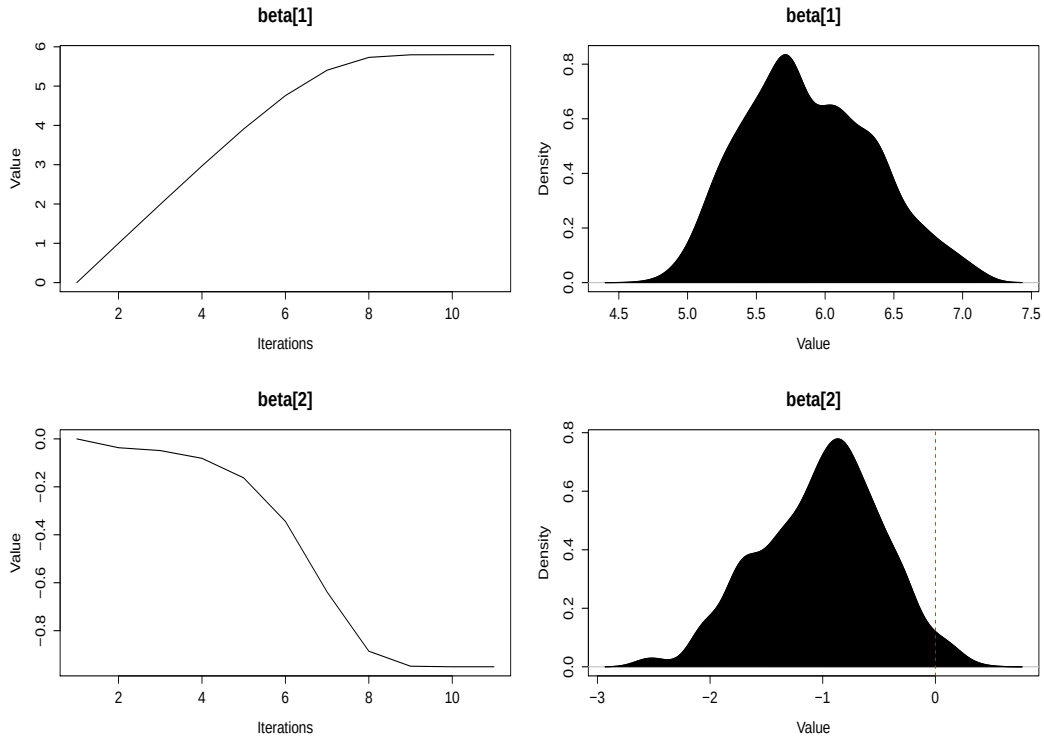

Figure 2.23: *Posterior density plots of regressor variables obtained by TR.*

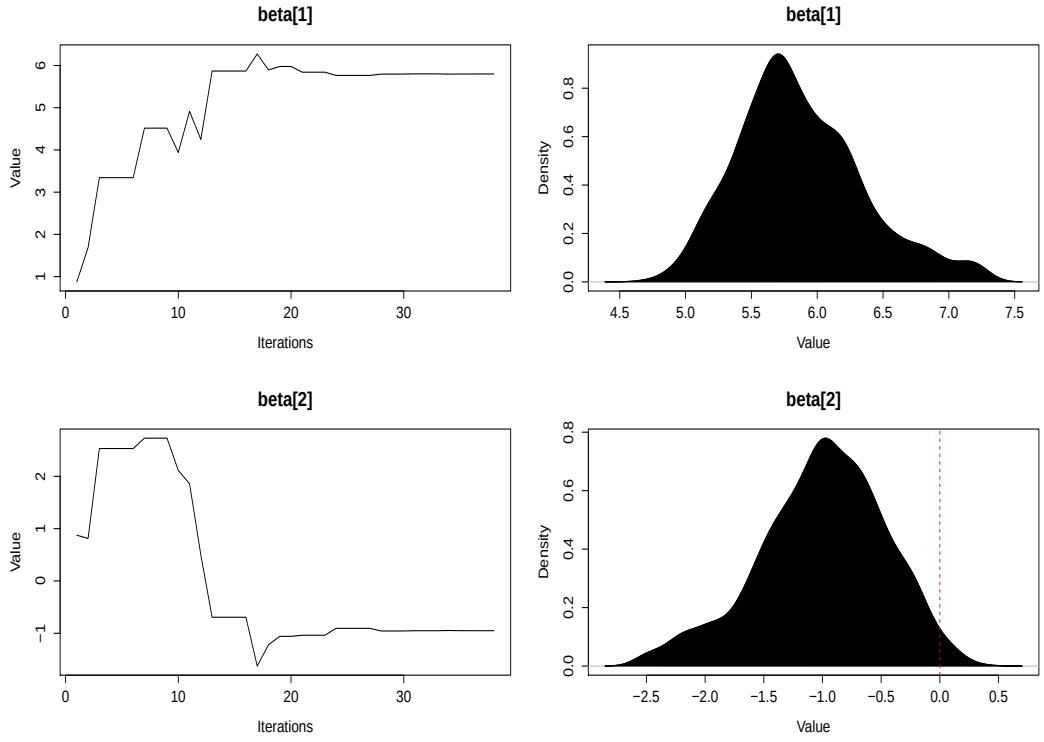

Figure 2.24: *Posterior density plots of regressor variables obtained by NM.*

Random-walk Metropolis algorithm							
	Mean	SD	MCSE	ESS	LB	Median	UB
beta[1]	5.85	0.46	0.04	185.725	5.03	5.82	6.88
beta[2]	-1.02	0.51	0.04	185.122	-2.17	-1.01	-0.05
Independent-Metropolis algorithm							
beta[1]	5.82	0.27	0.00	681.006	5.31	5.81	6.35
beta[2]	-0.97	0.30	0.00	711.345	-1.55	-0.96	-0.38

Table 2.12: *Simulated posterior summaries obtained by random walk and independent Metropolis algorithm under the assumption of exponential distribution.*

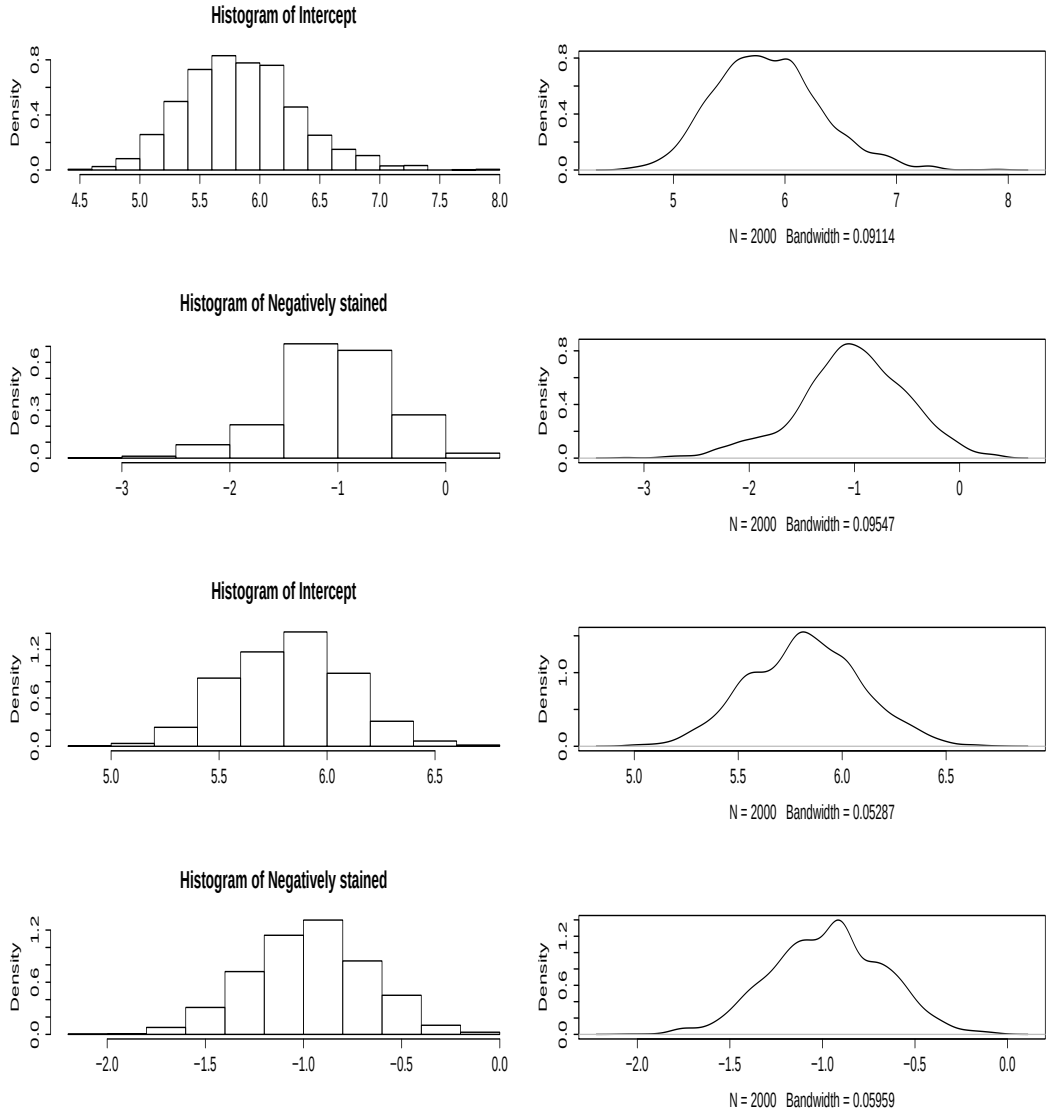

Figure 2.25: *Posterior density plots of regressor variables obtained by random walk and independent Metropolis algorithm.*

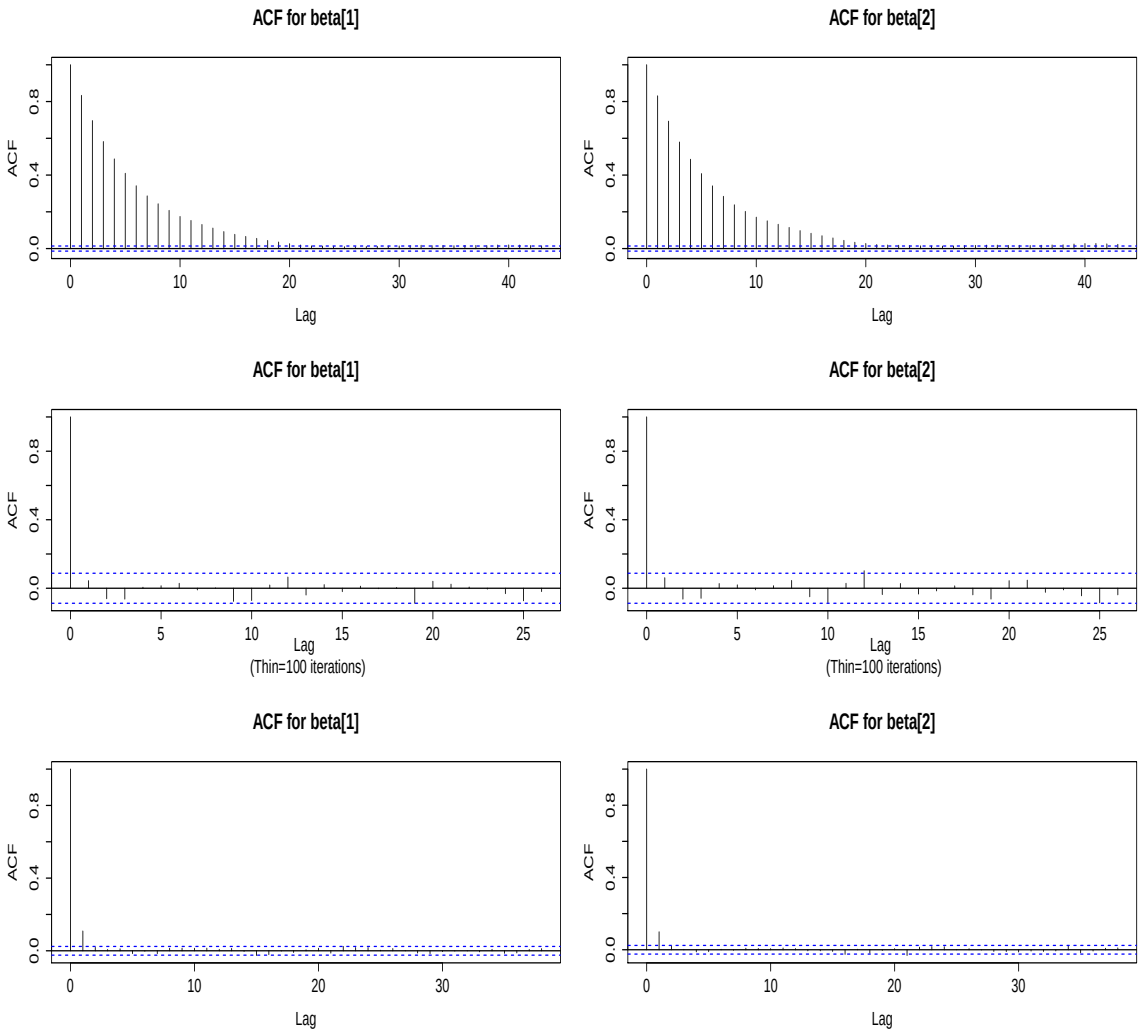

Figure 2.26: *Auto-correlations plots for exponential regression parameters beta1 and beta2 of breast cancer data using random-walk (uppermost and middlemost) and independent Metropolis algorithm (bottommost).*

2.6.1 Median and percentiles of exponential distribution

percentile of exponential distribution is given as

$$t_{med} = -\lambda \log\left(\frac{100-p}{100}\right)$$

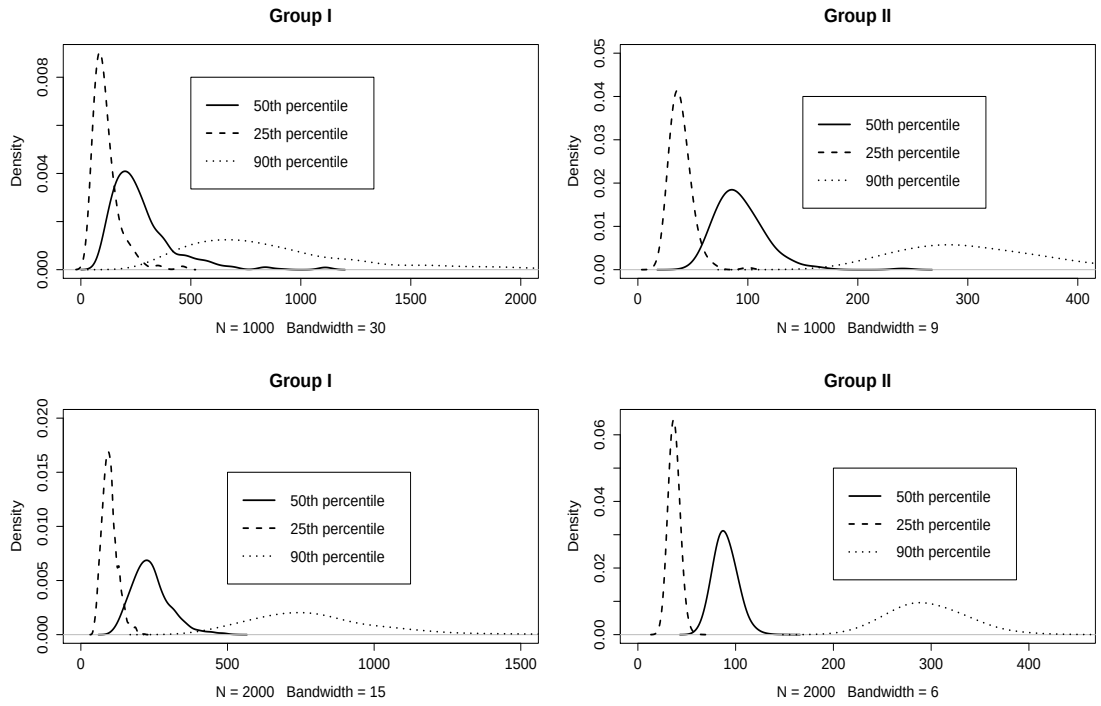

Figure 2.27: *Simulated posterior density plots of median and other percentile for both groups by SIR and IM algorithm.*

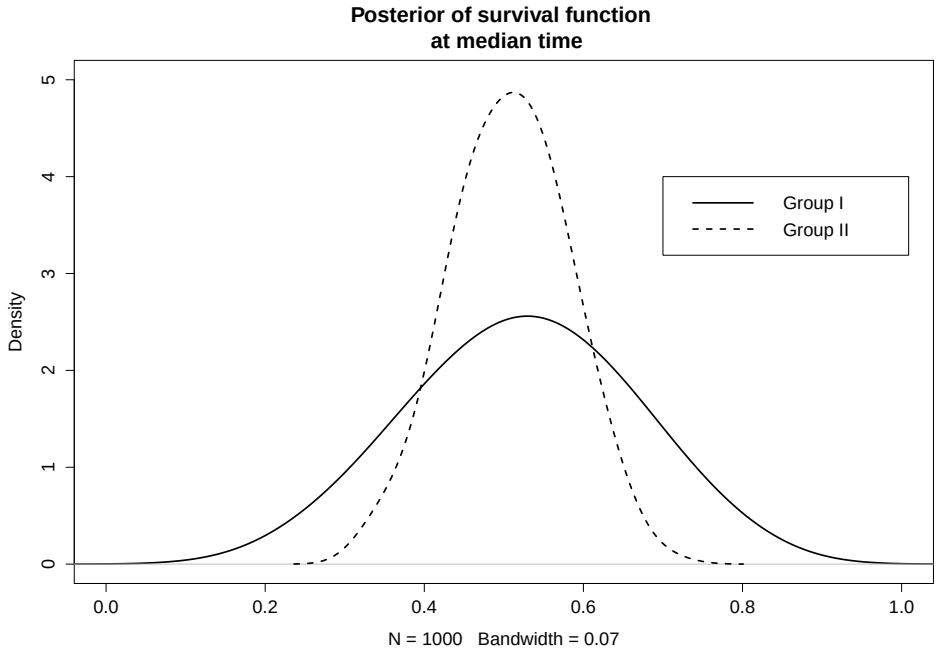

Figure 2.28: *Posterior density plots of survival function at* $t_{med} = 229$ *for group 1 and* $t_{med} = 88$ *for group 2.*

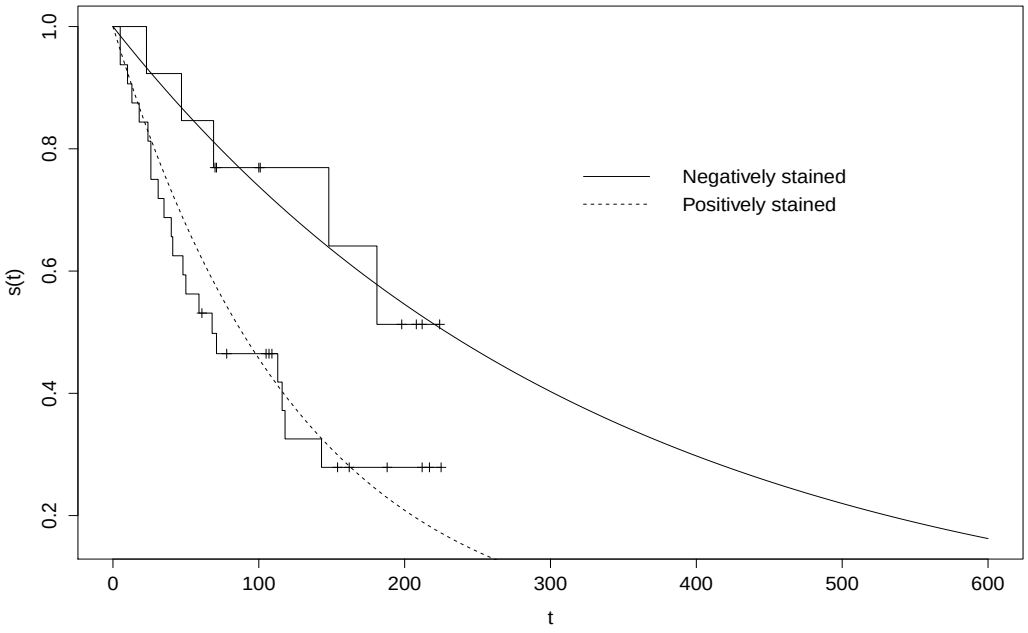

Figure 2.29: *Comparison of Kaplan-Meier survival curves with exponential survival curves.*

2.7 Model comparison

In this section, a goodness-of-fit criterion tests would be applied in order to verify which distribution fits better for this data. To compare the three models; namely generalized exponential, exponential extension and exponential distribution, the model selection criterion preferred by the Bayesians and likelihoodists are deviance and deviance information criterion (DIC, Spiegelhalter et al. 2002, is a model assessment tool). A smaller DIC and deviance indicates a better fit to the data set.

Models	Deviance	DIC
Generalized exponetial	251	253
Exponential extention	252	254
Exponential	314	315

Table 2.13: *Model comparison of generalized exponential, exponential extension and exponential model for breast cancer data. Both deviance and DIC criterion support generalized exponential distribution is a better choice as compared to exponential extension and exponential distribution. However, the difference between generalized exponential and exponential extension is magical. Contrary to this, the goodness of fit of these two models is much better than exponential.*

2.8 Conclusion

In this chapter the problem of discriminating between three distribution functions, namely exponential, generalized exponential and exponential extension are considered. All simulated numerical posterior summaries have been carried out using R software. The quantity e^{β} is the ratio of the hazard function for a woman with $X = 0$ (negatively stained)to that for a woman with $X = 1$ (positively stained), so that β is the logarithm of the ratio of the hazard of death at time t for negatively stained relative to positively stained women. For generalized exponential distribution the estimated value of this hazard ratio is $e^{-.95} = 0.38$.

$$\frac{\mathrm{p}(X = 0)}{\mathrm{p}(X = 1)} = .38$$

which gives,

$$\mathrm{p}(X = 1) = 2.55\ \mathrm{p}(X = 0)$$

Since this is greater than unity, we conclude that a woman who has a positively stained tumour will have approximately 3 time greater risk of death at any given time than a comparable women whose tumour was negatively stained. Positive staining therefore indicates a poorer prognosis for a breast cancer patient. For exponential extension and exponential distribution the hazard ratio is almost equal as for the generalized exponential distribution. Also the approximated and

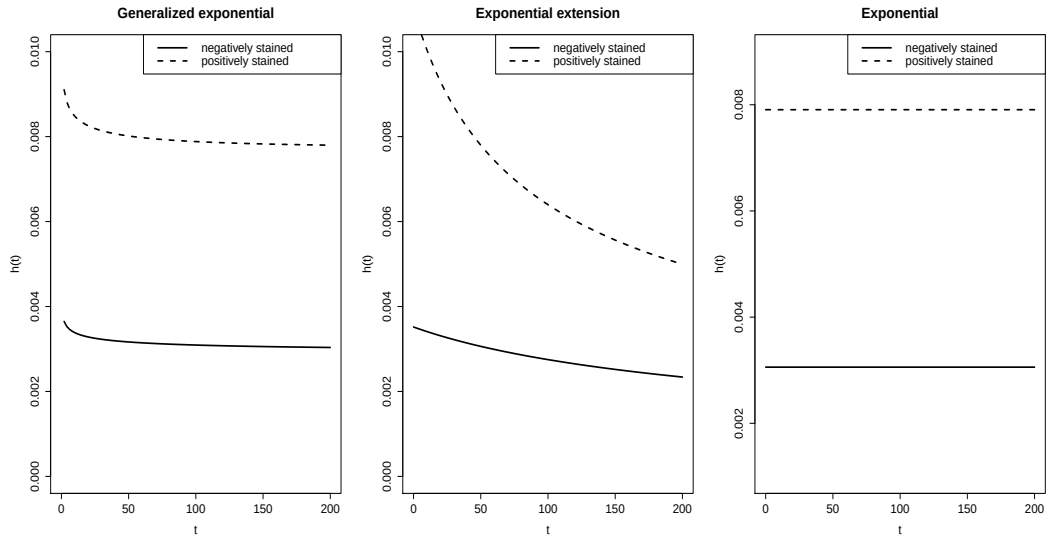

Figure 2.30: *hazard function of GE, EE and exponential distribution*

simulated median survival time for a women with negatively stained $(x_i = 0)$ is 247 days, while that for women with positively stained $(x_i = 1)$ is 95 days. The median survival for women with positively stained tumours is therefore about one third that of those whose tumours were negatively stained. It is also evident from

Figure 2.12, 2.22 and 2.29. These pictures shows the estimated survival curve for negatively and positively stained tumour. The survival time of negatively stained tumour is more as compared to positively stained tumour. The hazard plots of all the three distribution (GE, EE and exponential distribution) could be seen in Figure 2.30. The failure rate of GE for women having positively stained is more than the women with negatively staining as positive staining curve is above the negatively stained curve. This pattern is similar for the other two distributions, showing poor prognosis of women having positively stained tumour. Table 2.13 represents the model comparison of distributions. It could be seen that, there is very small difference in deviance and DIC values for generalized exponential (deviance= **251**, DIC = **253**) and exponential extension models (deviance= **252**, DIC = **254**) and a very large difference for exponential model (deviance= **314**, DIC = **315**). So it is clear that an addition of parameters makes the resulting distribution richer and more flexible for modelling data. Thus, it is justified that the use of GE and EE is appropriate for the given data. Hence, we can say that these two models could be a better choice as compared to exponential distribution for the analysis of survival data.

The use of Laplace approximation method made a great contribution in Bayesian framework. However, being an asymptotic approach one of the limitations of this approach is that this method is recommended for the data whose sample size is at least **5** times of the number of parameters available in a particular statistical model and it has been found in the present study that this approximation works well.

CHAPTER 3

Bayesian Regression Analysis of Exponentiated Weibull Lifetime Model

3.1 Introduction

In recent years the two parameter Weibull distribution has been most popular model for analyzing lifetime data (Murthy et al. 2004; Rinne 2009). Since the survival function and failure rate function of Weibull distribution has been obtained in a closed forms, it would be an easy task for modeling survival data. With this flexibility and advantage, its major weakness is its inability to accommodate non-monotone hazard rates (in particular, bathtub shaped hazard rates). This has lead to the need to seek generalizations of the Weibull distribution. A new family of distribution, namely "the exponentiated Weibull distribution (EW)" was first introduced by Mudholkar and Srivastava (1993) and Mudholkar et al. (1995), allowing for non-monotone hazard rates including the bathtub shaped hazard rate.

The two papers, Mudholkar and Srivastava (1993) and Mudholkar et al. (1995), have been most seminal in that they have initiated the development of distributions accommodating nonmonotone hazard rates (in particular, bathtub shaped hazard rates). In fact, since the publication of the two papers, many distributions and in particular other generalizations of the two-parameter Weibull distribution have been proposed, each allowing for nonmonotone and bathtub shaped hazard rates. The last few years have seen a surge of new generalizations proposed, mostly by statisticians in Brazil. Properties of EW have been studies in more detail by

Mudholkar and Hutson (1996), Jiang and Murthy (1999), Nassar and Eissa (2003). These authors have presented useful applications of the distribution in the modeling of flood data. Practically, the failure model of EW is more realistic than that of monotone failure rates and plays an important role to represent such data.

Currently, there are little studies for the use of the EW in reliability estimation. Singh et al. (2002, 2005a, 2005b and 2006) obtained Bayes estimations of the distribution parameters, reliability function and hazard function with type II censored sample under squared error as well as under LINEX loss function. Nassar and Eissa (2004) obtained the Bayes estimates of the two unknown parameters, the reliability and failure rate function by using Bayes approximation form due to Lindley (1980) under the squared error loss and LINEX loss functions. Elshahat (2006) derived Bayes estimators for the two unknown shape parameters of the EW based on progressive type I interval censored sample. Ashour and Afify (2007) considered the analysis of EW family distributed lifetime data observed under type I progressive interval censoring with random removals, maximum likelihood estimators of the parameters and their asymptotic variances are derived. Approximate Bayes estimators for the two unknown shape parameters are derived by Elshahat (2008) based on Lindley (1980) and Tierney and Kadane (1986). Approximate credible intervals for the unknown parameters are obtained with progressive interval censoring. Ashour and Afify (2008) derived maximum likelihood estimators for the parameters of EW with type II progressive interval censoring with random removals and their asymptotic variances. Kim et al. (2009) derived the maximum likelihood and Bayes estimators for EW lifetime model using symmetric and asymmetric loss functions.

Adding parameters to a well established family of distributions is a useful method for obtaining new families of distributions. Marshall and Olkin(1997) introduced an interesting approach of adding additional parameter to an existing model.

In this chapter, Bayesian analysis of three parameter exponentiated Weibull distribution is considered when all parameters are unknown. The four sub-models i.e Weibull, exponentiated exponential, Burr type X and Rayleigh distribution of EW distribution have also taken into consideration in Bayesian framework. For regression coefficient vector normal prior with mean zero and higher standard deviation is used, whereas for positive parameters half-Cauchy with scale **25** is used. The reason behind using these priors is to keep weak informative and proper priors for the parameters.

3.2 The model

A random variable T is said to have the EW distribution if its probability density function (pdf) and cumulative distribution function (cdf) are given by

$$f(t) = \gamma \frac{\alpha}{\lambda}\left(\frac{t}{\lambda}\right)^{\alpha-1} exp\left[-\left(\frac{t}{\lambda}\right)^{\alpha}\right]\left\{1-exp\left[-\left(\frac{t}{\lambda}\right)^{\alpha}\right]\right\}^{\gamma-1} \qquad t>0, \quad \gamma,\alpha,\lambda>0 \tag{3.1}$$

and

$$F(t) = \left\{1-exp\left[-\left(\frac{t}{\lambda}\right)^{\alpha}\right]\right\}^{\gamma} \tag{3.2}$$

where, γ and α are the shape parameters whereas λ is scale parameter. The particular case for $\alpha = 1$ is the exponentiated exponential (EE) distribution due to Gupta and Kundu (1999). The particular case for $\gamma = 1$ is the Weibull distribution. The particular case for $\alpha = 2$ is the Burr type X distribution studied by Sartawi and Abu-Salih (1991), Raqab (1998), Ahmad (2001), Mousa (2001), Jaheen and Al-Matrafi (2002), Kundu and Gupta (2004), Surles and D' Ambrosio (2004), Kundu and Raqab (2005, 2007), Malinowska and Szynal (2005), Surles and Padgett (2005), Raqab and Kundu (2006), Aludaat et al. (2008), Zhou et al. (2008), Alshunnar et al. (2010), and Montazer and Shayib (2010) among others. The particular case for $\gamma = 2$ and $\alpha = 1$ is the Rayleigh distribution.

The survival and hazard rate function of EW distribution are

$$S(t) = 1-\left\{1-exp\left[-\left(\frac{t}{\lambda}\right)^{\alpha}\right]\right\}^{\gamma} \tag{3.3}$$

and

$$h(t) = \frac{f(t)}{S(t)} \tag{3.4}$$

respectively. The hazard rate function allows for constant, monotonically increasing, monotonically decreasing, unimodal and bathtub shaped hazard rates. In particular, bathtub shapes with a unique change point occur when $\alpha > 1$ and $\gamma\alpha < 1$. Unimodal shapes with a unique change point occur when $\alpha < 1$ and $\gamma\alpha > 1$. Monotonically increasing shapes occur when $\alpha > 1$ and $\gamma\alpha > 1$. Monotonically decreasing shapes occur when $\alpha < 1$ and $\gamma\alpha < 1$. The hazard rate is constant when $\gamma = \alpha = 1$. Nadarajah (2009) compiles a collection of distributions allowing for bathtub hazard rates. Both the lower and upper tails of the hazard rate function behave polynomially.

The failure rates of exponentiated Weibull distribution is listed in the Table 3.1

3.2.1 Functions for exponentiated Weibull distribution in R

1. R code for probability density function is

Parameters	shape of exponentiated Weibull
$\alpha > 1$ and $\gamma\alpha < 1$	Bathtub shape with unique change point
$\alpha < 1$ and $\gamma\alpha > 1$	Unimodal shapes
$\alpha > 1$ and $\gamma\alpha > 1$	Monotonically increasing
$\alpha < 1$ and $\gamma\alpha < 1$	Monotonically decreasing
$\alpha = \gamma = 1$	Constant

Table 3.1: *Failure rates of exponentiated Weibull distribution.*

```
dexpweib<-function(x,alpha,gamma,lambda){
d1<-gamma*(alpha/lambda)*(x/lambda)^(alpha-1)
d2<-exp(-(x/lambda)^alpha)
d3<-(1-exp(-(x/lambda)^alpha))^(gamma-1)
d<-(d1*d2*d3)
return(d)
}
```

2. R code for cumulative density function is

```
pexpweib<-function(x,alpha,gamma,lambda){
p<-(1-exp(-(x/lambda)^alpha))^(gamma)
return(p)
}
```

3. R code for random generation function is

```
rexpweib<-function(n,alpha,gamma,lambda){
u<-runif(n)
x<--1/lambda*(log(1-(u)^(1/alpha)))^1/gamma
return(x)
}
```

4. R code for survival function is

```
sexpweib<-function(x,alpha,gamma,lambda){
surv<-(1-pexpweib(x,alpha,gamma,lambda))
return(surv)
}
```

5. R code for hazard function is

```
hexpweib<-function(x,alpha,gamma,lambda){
haz<-dexpweib(x,alpha,gamma,lambda)/sexpweib(x,alpha,gamma,lambda)
return(haz)
}
```

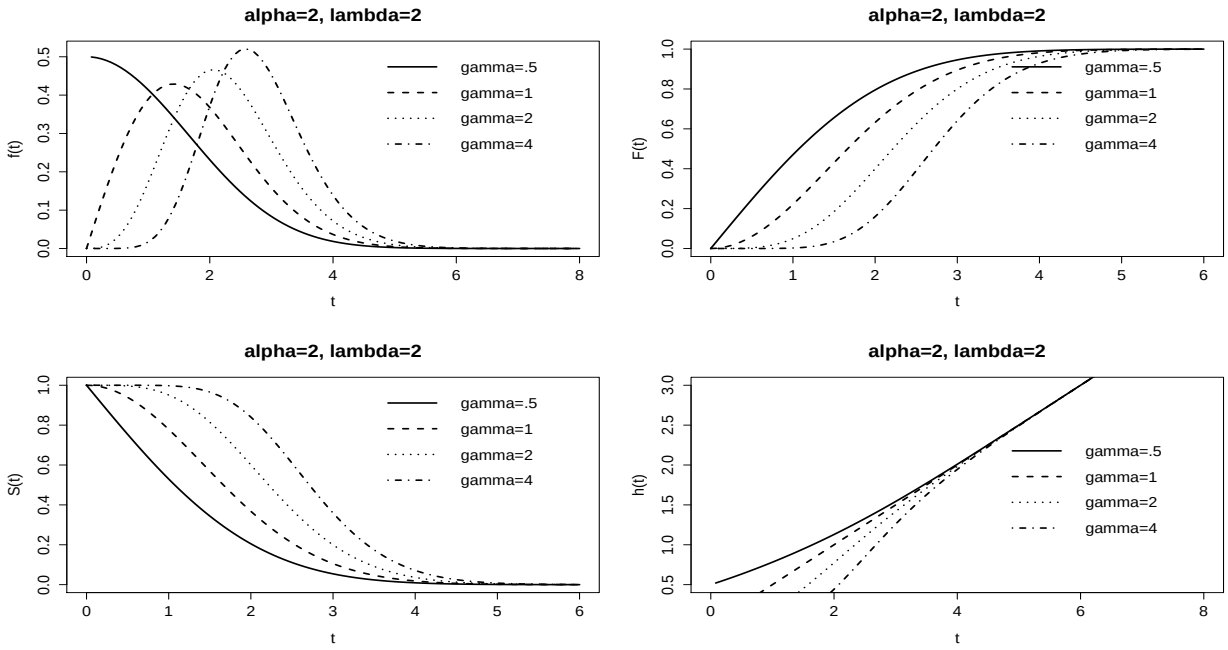

Figure 3.1: Showing the pdf, cdf, survival and failure rate curve of EW distribution for $\alpha = 2, \lambda = 2,$ when $\gamma = .5, 1, 2, 4$

The first plot in the uppermost panel of Figure 3.1 shows that the density function of EW is unimodal and, for fixed λ and α, it becomes more and more symmetric as γ increases. Plot in the second panel of the first row shows the cumulative density curve. The first plot in the second row of Figure 3.1 is the survival curve and the hazard plot in the second row and second panel shows that the failure rate is a non-decreasing function of γ for fixed λ and α.

3.3 Exponentiated Weibull model and its sub-models

In this section, discussion regarding the analysis of EW model in Bayesian environment is considered. Let $(t_1,\delta_1),(t_2,\delta_2)\ldots,(t_n,\delta_n)$ be a random sample of size n of lifetimes generated by an EW distribution with parameters α, γ and λ and that $(\delta_i = 1)$ if t_i is completely observed or $(\delta_i = 0)$ if t_{c_i} is a right censored observation $(i = 1,\ldots,n)$. The t's are assumed to be independent and identically distributed with density $f(t)$ and survival function $S(t)$. The exact survival time t_i of an individual

will be observed only if $t_i \leq t_{c_i}$. The data in this framework can be represented by the n pairs of random variables (y_i, δ_i), where

$$y_i = min(t_i, t_{c_i})$$

and

$$\delta_i = \begin{cases} 1 & \text{if } t_i \leq t_{c_i}, \\ 0 & \text{if } t_i > t_{c_i}. \end{cases} \tag{3.5}$$

Assuming a non-informative censoring mechanism, Lawless (2003), the likelihood and log-likelihood functions are given, respectively, by:

$$L \propto \prod_{i=1}^{n} [f(y_i)]^{\delta_i} [S(t_{c_i})]^{1-\delta_i}$$

using Equation 3.1 and 3.3, the likelihood function is given by,

$$L(y;\alpha,\gamma,\lambda) \propto \prod_{i=1}^{n} \left[\left(\gamma \frac{\alpha}{\lambda} \left(\frac{y_i}{\lambda} \right)^{\alpha-1} exp\left(-\left(\frac{y_i}{\lambda} \right)^{\alpha} \right) \left(1 - exp\left(-\left(\frac{y_i}{\lambda} \right)^{\alpha} \right) \right)^{\gamma-1} \right)^{\delta_i} \right.$$
$$\left. \left(1 - \left\{ 1 - exp\left(-\left(\frac{t_{c_i}}{\lambda} \right)^{\alpha} \right) \right\}^{\gamma} \right)^{1-\delta_i} \right] \tag{3.6}$$

Taking natural logarithm, we get

$$\log L(t;\alpha,\gamma,\lambda) \propto \sum_{i=1}^{n} \delta_i \log \left[\gamma \frac{\alpha}{\lambda} \left(\frac{y_i}{\lambda} \right)^{\alpha-1} exp\left(-\left(\frac{y_i}{\lambda} \right)^{\alpha} \right) \left(1 - exp\left(-\left(\frac{y_i}{\lambda} \right)^{\alpha} \right) \right)^{\gamma-1} \right] +$$
$$\sum_{i=1}^{n} (1-\delta_i) \log \left[1 - \left\{ 1 - exp\left(-\left(\frac{t_{c_i}}{\lambda} \right)^{\alpha} \right) \right\}^{\gamma} \right] \tag{3.7}$$

Now, consider non-informative priors for the parameters α,

$$\alpha \sim \text{half-Cauchy}(\sigma),$$

and for γ

$$\gamma \sim \text{half-Cauchy}(\sigma).$$

which gives,

$$p(\alpha) = \frac{2\sigma}{\pi(\alpha^2 + \sigma^2)},$$

and

$$p(\gamma) = \frac{2\sigma}{\pi(\gamma^2 + \sigma^2)}.$$

respectively. Since, $\lambda > 0$ and β can take any value on the real line, hence, log link function is used

$$log(\lambda) = \mathbf{X}\beta$$

where, $\boldsymbol{X}$ is model matrix and $\boldsymbol{\beta}$ is the vector of regression coefficients,
or, equivalently,

$$\lambda = e^{\mathbf{X}\beta}$$

$$\beta_j \sim N(0, 1000)$$

Combining the above equations with Equation 3.7 and using Bayes theorem, the joint posterior distribution is derived as follows

$$\begin{aligned} p(\alpha,\gamma,\beta|y,X) \propto \prod_{i=1}^{n} \Bigg[\Bigg(\gamma \frac{\alpha}{\lambda} \left(\frac{y_i}{\lambda}\right)^{\alpha-1} exp\left(-\left(\frac{y_i}{\lambda}\right)^{\alpha}\right) \left(1 - exp\left(-\left(\frac{y_i}{\lambda}\right)^{\alpha}\right)\right)^{\gamma-1} \Bigg)^{\delta_i} \\ \left(1 - \left\{1 - exp\left(-\left(\frac{t_{c_i}}{\lambda}\right)^{\alpha}\right)\right\}^{\gamma}\right)^{1-\delta_i} \Bigg] \times \left\{\frac{2\sigma}{\pi(\alpha^2+\sigma^2)}\right\} \\ \times \left\{\frac{2\sigma}{\pi(\gamma^2+\sigma^2)}\right\} \cdot \prod_{j=1}^{p} \left\{\frac{1}{\sqrt{2\pi}1000} e^{\frac{-\beta_j^2}{2*1000^2}}\right\} \end{aligned} \tag{3.8}$$

Marginal posterior of unknown parameters is obtained by integrating the joint posterior distribution with respect to the other parameters. Hence, the marginal posterior of $\alpha, \gamma,$ and β can be written as
marginal posterior of α

$$p(\alpha|t,X) = \int_0^{\infty} \int_{-\infty}^{\infty} p(\alpha,\gamma,\beta|t,X)\, d\beta\, d\gamma \tag{3.9}$$

marginal posterior of β

$$p(\beta|t,X) = \int_0^{\infty} \int_0^{\infty} p(\alpha,\gamma,\beta|t,X)\, d\alpha\, d\gamma \tag{3.10}$$

marginal posterior of γ

$$p(\gamma|t,X) = \int_{-\infty}^{\infty} \int_0^{\infty} p(\alpha,\gamma,\beta|t,X)\, d\beta\, d\alpha \tag{3.11}$$

3.3.1 Sub-models of exponentiated Weibull model

There are four important sub-models:

1. $\alpha = 1$ gives exponentiated exponential distribution
2. $\gamma = 1$ gives the weibull distribution
3. $\alpha = 2, \gamma = 1$ gives the Rayleigh distribution
4. $\alpha = 2$ gives Burr type X distribution

These four sub-models will also be discussed in Bayesian scenario and at the end of chapter a comparison has also been made with EW distribution.

3.3.2 Exponentiated exponential model

Exponentiated exponential lifetime model has been first introduced by Gupta and Kundu (1999). The pdf and cdf are given as

$$f(t) = \frac{\gamma}{\lambda} exp\left\{-\left(\frac{t}{\lambda}\right)\right\}\left[1 - exp\left(-\left(\frac{t}{\lambda}\right)\right)\right]^{\gamma-1} \qquad t > 0, \quad \alpha, \lambda > 0 \tag{3.12}$$

and

$$F(t) = \left[1 - exp\left(-\left(\frac{t}{\lambda}\right)\right)\right]^{\gamma} \tag{3.13}$$

The survival and hazard rate function of exponentiated exponential distribution are

$$S(t) = 1 - \left[1 - exp\left(-\left(\frac{t}{\lambda}\right)\right)\right]^{\gamma} \tag{3.14}$$

and

$$h(t) = \frac{\frac{\gamma}{\lambda} exp\left[-\left(\frac{t}{\lambda}\right)\right]}{1 - exp\left[-\left(\frac{t}{\lambda}\right)\right]} \tag{3.15}$$

respectively.

3.3.3 Functions for exponentiated exponential distribution in R

1. R code for probability density function is

```
dexpoexp<-function(x,gamma,lambda){
d1<-(gamma/lambda)*exp(-(x/lambda))
d2<-(1-exp(-(x/lambda)))^(gamma-1)
d<-(d1*d2)
return(d)
}
```

2. R code for cumulative density function is

```
pexpoexp<-function(x,gamma,lambda){
p<-(1-exp(-(x/lambda)))^(gamma)
return(p)
}
```

3. R code for random generation function is

```
rexpoexp<-function(n,shape,scale){
u<-runif(n)
t<--scale*log(1-(u)^(1/shape))
}
```

4. R code for survival function is

```
sexpoexp<-function(x,gamma,lambda){
surv<-(1-pexpweib(x,gamma,lambda))
return(surv)
}
```

5. R code for hazard function is

```
hexpoexp<-function(x,gamma,lambda){
haz<-dexpoexp(x,gamma,lambda)/sexpoexp(x,gamma,lambda)
return(haz)
}
```

The joint posterior distribution of exponentiated exponential model

$$p(\gamma,\beta|y,X) \propto \prod_{i=1}^{n} \left[\left\{ \frac{\gamma}{e^{x_i\beta}} exp\left\{ -\left(\frac{y_i}{e^{x_i\beta}} \right) \right\} \left[1 - exp\left(-\left(\frac{y_i}{e^{x_i\beta}} \right) \right) \right]^{\gamma-1} \right\}^{\delta_i} \left\{ \left[1 - \left[1 - exp\left(-\left(\frac{t_{c_i}}{e^{x_i\beta}} \right) \right) \right]^{\gamma} \right] \right\}^{(1-\delta_i)} \right] . \left\{ \frac{2\sigma}{\pi(\gamma^2+\sigma^2)} \right\} \times \prod_{j=1}^{p} \left\{ \frac{1}{\sqrt{2\pi}1000} e^{\frac{-\beta_j^2}{2*1000^2}} \right\} \tag{3.16}$$

3.3.4 The Burr type X lifetime model

Burr (1942) introduced twelve different forms of cumulative distribution functions for modelling data. Among those twelve distribution functions, Burr-Type X has received the maximum attention. Two-parameter Burr-Type X distribution is particular case of EW model for $\alpha = 2$, has the following cumulative distribution function (CDF);

$$F(t) = \left\{ 1 - exp\left(-\left(\frac{t}{\lambda} \right)^2 \right) \right\}^{\gamma} \tag{3.17}$$

Therefore, the Burr-type X has the density function for t > 0 as;

$$f(t) = 2\left(\frac{\gamma}{\lambda}\right)\left(\frac{t}{\lambda}\right) exp\left[-\left(\frac{t}{\lambda}\right)^2\right]\left\{1 - exp\left(-\left(\frac{t}{\lambda}\right)^2\right)\right\}^{\gamma-1} \qquad t > 0, \quad \alpha, \lambda > 0 \tag{3.18}$$

3.3.5 Functions for Burr type X distribution in R

1. R code for probability density function is

```
dburr<-function(x,gamma,lambda){
d1<-2*(gamma/lambda)*(x/lambda)
d2<-exp(-(x/lambda)^2)
d3<-(1-exp(-(x/lambda)^2))^(gamma-1)
d<-(d1*d2*d3)
return(d)
}
```

2. R code for cumulative density function is

```
pburr<-function(x,gamma,lambda){
p<-(1-exp(-(x/lambda)^2))^(gamma)
return(p)
}
```

3. R code for random generation function is

```
rburr<-function(n,shape,scale){
u<-runif(n)
t<--scale*(log(1-(u)^(1/shape)))^1/2
}
```

4. R code for survival function is

```
sburr<-function(x,gamma,lambda){
surv<-(1-pburr(x,gamma,lambda))
return(surv)
}
```

5. R code for hazard function is

```
hburr<-function(x,gamma,lambda){
haz<-dburr(x,gamma,lambda)/sburr(x,gamma,lambda)
return(haz)
}
```

The joint posterior distribution is derived as follows

$$p(\gamma,\beta|y,X) \propto \prod_{i=1}^{n} \left[\left\{ 2\left(\frac{\gamma}{e^{x_i\beta}}\right)\left(\frac{y_i}{e^{x_i\beta}}\right) exp\left[-\left(\frac{y_i}{e^{x_i\beta}}\right)^2\right] \left\{1-exp\left(-\left(\frac{y_i}{e^{x_i\beta}}\right)^2\right)\right\}^{\gamma-1} \right\}^{\delta_i} \right.$$
$$\left. \left\{ \left[1-\left\{1-exp\left(-\left(\frac{t_{c_i}}{e^{x_i\beta}}\right)^2\right)\right\}^{\gamma} \right] \right\}^{(1-\delta_i)} \right] . \left\{ \frac{2\sigma}{\pi(\gamma^2+\sigma^2)} \right\}$$
$$\times \prod_{j=1}^{p} \left\{ \frac{1}{\sqrt{2\pi}1000} e^{\frac{-\beta_j^2}{2*1000^2}} \right\} \tag{3.19}$$

3.3.6 The Rayleigh lifetime model

The Rayleigh distribution was originally introduced by Lord Rayleigh (1880) in the field of acoustics. Also, it has a wide application in lifetime data analysis especially in reliability theory and survival analysis. Siddiqui (1962) discussed the origin and properties of the Rayleigh distribution. Inference for model Rayleigh model has been considered by Sinha and Howlader (1983) and Lalitha and Mishra (1996). An important characteristic of the Rayleigh distribution is that its hazard function is an increasing function of time. The pdf and cdf of Rayleigh distribution is given by

$$f(t) = 2\left(\frac{1}{\lambda}\right)\left(\frac{t}{\lambda}\right) exp\left(-\left(\frac{t}{\lambda}\right)^2\right) \tag{3.20}$$

$$F(t) = 1-exp\left(-\left(\frac{t}{\lambda}\right)^2\right) \tag{3.21}$$

Survival

$$S(t) = exp\left(-\left(\frac{t}{\lambda}\right)^2\right) \tag{3.22}$$

and hazard function

$$h(t) = 2\left(\frac{1}{\lambda}\right)\left(\frac{t}{\lambda}\right) \tag{3.23}$$

3.3.7 Functions for Rayleigh distribution in R

1. R code for probability density function is

```
dray<-function(x,lambda){
d1<-2*(1/lambda)*(x/lambda)
d2<-exp(-(x/lambda)^2)
d<-(d1*d2)
return(d)
}
```

2. R code for cumulative density function is

```
pray<-function(x,lambda){
p<-(1-exp(-(x/lambda)^2))
return(p)
}
```

3. R code for random generation function is

```
rray<-function(n,scale){
u<-runif(n)
t<--scale*(log(1-u))^1/2
}
```

4. R code for survival function is

```
sray<-function(x,lambda){
surv<-(1-pburr(x,lambda))
return(surv)
}
```

5. R code for hazard function is

```
hray<-function(x,lambda){
haz<-dburr(x,lambda)/sburr(x,lambda)
return(haz)
}
```

The joint posterior distribution of Rayleigh distribution is

$$p(\beta|y,X) \propto \prod_{i=1}^{n} \left[\left\{ 2\left(\frac{1}{e^{x_i\beta}}\right)\left(\frac{y_i}{e^{x_i\beta}}\right) exp\left(-\left(\frac{y_i}{e^{x_i\beta}}\right)^2\right)\right\}^{\delta_i} \left\{ exp\left(-\left(\frac{t_{c_i}}{e^{x_i\beta}}\right)^2\right)\right\}^{(1-\delta_i)} \right].$$

$$\times \prod_{j=1}^{p} \left\{ \frac{1}{\sqrt{2\pi}1000} e^{\frac{-\beta_j^2}{2*1000^2}} \right\} \tag{3.24}$$

3.3.8 Weibull lifetime model

The Weibull model is very flexible and has been found to provide a good description of many types of time-to-event data. This is the distribution which occupy a central role because of its demonstrated usefulness in a wide range of situations. There are many potential life time models but this model is used quite effectively to analyze skewed data sets. Weibull distribution has two parameter, shape and scale. Its density, survival and hazard functions, respectively are:

$$f(t) = \frac{\alpha}{\lambda}\left(\frac{t}{\lambda}\right)^{\alpha-1} exp\left[-\left(\frac{t}{\lambda}\right)^{\alpha}\right]$$

$$S(t) = exp\left[-\left(\frac{t}{\lambda}\right)^{\alpha}\right]$$

$$h(t) = \frac{\alpha}{\lambda}\left(\frac{t}{\lambda}\right)^{\alpha-1}$$

where $\lambda > 0$ and $\gamma > 0$ are the rate and shape parameters respectively. Papers based on Bayesian analysis of Weibull model are Khan and Khan (2013a, 2013b, 2014) and Khan et al. (2015). Since, we have availability of distribution function for Weibull distribution in `R` so we do not need to define this distribution as done for previous models. To get the probability density function of Weibull distribution, simply go to the `R` prompt and type

```
help(dweibull)
```

Then a web page regarding full description of probability density function of Weibull distribution will open. The arguments of `dweibull` is

```
dweibull(x, shape, scale = 1, log = FALSE)
```

For cumulative distribution function,

```
help(pweibull)
pweibull(q, shape, scale = 1, lower.tail = TRUE, log.p = FALSE)
```

for random generation function,

```
help(rweibull)
rweibull(n, shape, scale = 1)
```

for survival function,

```
surv<-function(x,shape,scale){
pweibull(x,shape,scale=1,lower.tail=FALSE,log.p=TRUE)
}
```

for hazard function,

```
haz<-function(x,shape,scale){
dweibull(x, shape, scale = 1, log = FALSE)/pweibull(x,
shape,scale=1,lower.tail=FALSE,log.p=TRUE)
}
```

The joint posterior distribution of Weibull distribution is

$$p(\alpha,\beta|y,X) \propto \prod_{i=1}^{n}\left[\left\{\frac{\alpha}{e^{x_i\beta}}\left(\frac{y_i}{e^{x_i\beta}}\right)^{\alpha-1} exp\left[-\left(\frac{y_i}{\lambda}\right)^{\alpha}\right]\right\}^{\delta_i}\left\{exp\left[-\left(\frac{t_{c_i}}{e^{x_i\beta}}\right)^{\alpha}\right]\right\}^{(1-\delta_i)}\right]. \times\left\{\frac{2\sigma}{\pi(\alpha^2+\sigma^2)}\right\}\times\prod_{j=1}^{p}\left\{\frac{1}{\sqrt{2\pi}1000}e^{\frac{-\beta_j^2}{2*1000^2}}\right\} \tag{3.25}$$

From Equations 3.8, 3.16, 3.19, 3.24 and 3.25 it is clear that it is not possible to get explicit forms for the marginal posterior distributions for each parameter. Consequently, some approximation method such as the Laplace approximation is required. When models become too difficult to be analyze analytically, we have to use simulation algorithms, such as the Markov chain Monte Carlo (MCMC) methods to obtain posterior estimates. The Markov chain Monte Carlo (MCMC) method is a general simulation method for sampling from posterior distributions and computing posterior quantities of interest. To simulate samples of the joint posterior distribution of interest, we need to sample successively from a target distribution. The Gibbs algorithm requires to decompose the joint posterior distribution into full conditional distributions for each parameter in the model and then sample from each one of these conditional distributions.

For the exponentiated Weibull distribution and its sub-models, the conditional posterior densities for α, γ and λ show that standard sampling schemes are not feasible since the conditional distributions are not given in a known form. In this way, an alternative target distribution to the full conditional distributions should

be used. The alternative proposal distribution should be a distribution from which it is easy to sample from it; in this way, we use Metropolis-Hastings algorithms. Tierney (1994) suggested, when possible, use of the Metropolis-Hastings algorithm within Gibbs sampling to sample from full conditional distributions.

In our applications, to sample from the full conditional distributions for α, γ and λ, we have used the Independent Metropolis algorithm (IM) Proposed by Hastings (1970) and popularized by Tierney (1994). The Independence Metropolis (IM) algorithm (also called the independence sampler) is an algorithm in which the proposal distribution does not depend on the previous state or iteration. The proposal distribution must be a good approximation of the target distribution for the IM algorithm to perform well, and the proposal distribution should have slightly heavier tails than the target distribution. IM is used most often to obtain additional posterior samples given an algorithm that has already converged. Since IM is nonadaptive and uses a proposal distribution that remains fixed for all iterations, it may be used as a final algorithm. The MCMC procedure provides a flexible environment for fitting a wide range of models. In this way, we should use some approximation method to solve integrals as the Laplace method and simulation tools such as Markov chain Monte Carlo.

3.4 Data set: Survival of multiple myeloma patients

Multiple myeloma (MM), a neoplasm of plasma cells, affects 1 to 5 per 100,000 individuals each year worldwide with a higher incidence in the West. It is the second most common hematologic malignancy in the United States, and it is estimated that there will be 19,900 new diagnoses and 10,790 deaths due to myeloma in 2007, Kumar et al. (2008). The data were obtained from Krall et al. (1975), related to **48** patients, all of whom were aged between **50** and **80** years. This data is also discussed by Collett (1994, 2003). Some of these patients had not died by the time that the study was completed, and so these individuals contribute right-censored survival times. The aim of the study carried out at the Medical Center of the University of West virginia, USA, was to examine the association between the values of certain explanatory variables. In the study, the primary response variable was the time, in months, from diagnosis until death from multiple myeloma. The variables in this data set are listed as

Variable	Variable name	coding
1	Age	Years (integer counts)
2	Sex	1= male, 0= female
3	Bun	Blood urea nitrogen
4	Ca	Serum calcium
5	Hb	Serum haemoglobin
6	Pcells	Percentage of plasma cells
7	Protein	Bence-Jones protein (0 = absent, 1 =present)

3.5 Bayesian modeling of exponentiated Weibull model

In this section an attempt has been made to implement Bayesian methods for censored data related to modelling of EW distribution. These implementation is done through `LaplacesDemon` package. However, this package does not have distribution function for EW distribution. For the computation of posterior density it is necessary to have probability density function and survival function. So, we have written functions for these statistical probabilities. Also, they have not discussed the implementation in the case of censored data, which is a primary requirement in survival analysis. In this section, functions to handle censored observation is created in `R`. The package in which these functions are called is `LaplacesDemon`. This package is having two important functions namely, `LaplaceApproximation` and `LaplacesDemon`. `LaplaceApproximation` is meant for implementation of asymptotic approximation theory and `LaplacesDemon` is for implementation of simulation tools. Fitting of multiple myeloma data in Bayesian aura includes the following `R` codes.

3.5.1 Creation of multiple myeloma data

For the purpose of demonstration, a real survival data of multiple myeloma patients has been taken for Bayesian modeling of exponentiated Weibull distribution. The data contains seven regressor variables namely, Age, Sex, Bun, Calcium, Haemoglobin, Pcells and Protein. The vector of these regressor variables has been define with an object names $x_1, x_2, x_3, x_4, x_5, x_6$ and x_7, respectively. The object `y` is the vector of lifetime of the patients which includes censoring time and a binary vector, called `censor`, has been created which contains 1 and 0, where 1 stands for observed lifetime and 0 stands for censored lifetime. The object X is called the model matrix which contains seven columns (seven regressor variables) and also a

column of $\mathbf{1}'s$ is also inserted into it as an intercept. Here $J = 8$, as X has eight columns.

```
library(LaplacesDemon)
y<-c(13,52,6,40,10,7,66,10,10,14,16,4,65,5,11,10,15,5,76,56,88,24,
   51,4,40,8,18,5,16,50,40,1,36,5,10,91,18,1,18,6,1,23,15,18,12,12
   ,17,3)

censor<-c(1,0,1,1,1,0,1,0,1,1,1,1,1,1,0,1,0,1,0,0,1,1,1,1,0,1,1,1,1,
        1,1,1,1,1,1,1,0,1,0,1,1,1,1,1,0,1,1,0)
#age
x1<-c(66,66,53,69,65,57,52,60,70,70,68,50,59,60,66,51,55,67,60,66,
    63,67,60,74,72,55,51,70,53,74,70,67,63,77,61,58,69,57,59,61,75,
    56,62,60,71,60,65,59)
x11<-x1-mean(x1)
#Sex
x2<-c(0,0,1,0,0,1,0,0,0,0,0,1,0,0,1,1,0,1,0,0,0,0,1,0,0,0,0,1,0,0,1,
    0,0,0,0,1,1,0,1,1,0,1,1,1,1,1,1,0)
#Bun
x3<-c(25,13,15,10,20,12,21,41,37,40,39,172,28,13,25,12,14,26,12,18,
    21,10,10,48,57,53,12,130,17,37,14,165,40,23,13,27,21,20,21,11,
    56,20,21,18,46,6,28,90)
x31<-x3-mean(x3)
#Ca
x4<-c(10,11,13,10,10,8,10,9,12,11,10,9,9,10,9,9,9,8,12,11,9,10,10,
    9,9,12,15,8,9,13,9,10,9,8,10,11,10,9,10,10,12,9,10,9,9,10,8,
    10)
x41<-x4-mean(x4)
#Hb
x5<-c(14.6,12,11.4,10.2,13.2,9.9,12.8,14,7.5,10.6,11.2,10.1,6.6,
    9.7,8.8,9.6,13,10.4,14,12.5,14,12.4,10.1,6.5,12.8,8.2,14.4,
    10.2,10,7.7,5,9.4,11,9,14,11,10.8,5.1,13,5.1,11.3,14.6,8.8,
    7.5,4.9,5.5,7.5,10.2)
x51<-x5-mean(x5)
#Pcells
x6<-c(18,100,33,30,66,45,11,70,47,27,41,46,66,25,23,80,8,49,9,90,
    42,44,45,54,28,55,100,23,28,11,22,90,16,29,19,26,33,100,100,
    100,18,3,5,85,62,25,8,6)
```

```
x61<-x6-mean(x6)
#Protein
x7<-c(1,0,1,1,0,0,1,1,0,0,0,1,0,0,0,0,0,0,0,0,1,0,1,0,1,0,0,0,0,
    1,0,0,1,0,0,1,0,1,0,0,0,0,0,1,0,0,0,1)
library(LaplacesDemon)
X<-cbind(1,x11,x2,x31,x41,x51,x61,x7)
J<-8
mon.names<-c("LP","shape1","shape2")
parm.names<-as.parm.names(list(beta=rep(0,J),
           log.shape1=0,log.shape2=0))
MyData<-list(J=J,X=X,mon.names=mon.names,parm.names=
        parm.names,y=y,censor=censor)
```

Each parameter must have a name specified in the vector `parm.names`, and parameter names must be included with the data. The object is created by using the function `as.parm.names`. The object `mon.names` is meant for the variables to be monitored. Object `MyData` has been created by making use of lists of vectors, that are, `J`, `X`, monitored variables `mon.names`, vector of parameters `parm.names`, vector of survival time which includes censored and uncensored observation `y` and an individual vector of censored observation using indicator variable, called `censor`.

3.5.2 R code for model specification of exponentiated Weibull distribution

Since exponentiated Weibull distribution has two shape and one scale parameter, so the function `Model` includes the definition of parameters as `shape1`, `shape2` and `beta` for vector of coefficients. Prior assigned for shape parameters is half Cauchy and for $\beta's$ is normal distribution. After defining parameters and its prior the next step is to construct the log likelihood of the distribution. For defining log likelihood function which contains censoring mechanism in it, we need to define density function and survival function of the distribution, which is defined in the function `Model` as f_1 and s_1, respectively. Then, the `Model` function is evaluated and the logarithm of the unnormalized joint posterior density is calculated as `LP`. This function returns an object called `Modelout`, which is a list of the objects, log-posterior (`LP`), deviance = -2*log-likelihood (`Dev`), monitored variable (`Monitor`), `yhat` and `parm`.

```
Model<-function(parm,Data)
```

```
{
beta<-parm[1:Data$J]
shape1<-exp(parm[Data$J+1])
shape2<-exp(parm[Data$J+2])
beta.prior<-sum(dnorm(beta,0,1000,log=T))
shape1.prior<-dhalfcauchy(shape1,25,log=T)
shape2.prior<-dhalfcauchy(shape2,25,log=T)
mu<-tcrossprod(beta,Data$X)
scale<-exp(mu)
f1<-function(y,shape1,shape2,scale) shape2*dweibull(y,shape=
    shape1,scale)*pweibull(y,shape=shape1,scale)^(shape2-1)
s1<-function(y,shape1,shape2,scale) 1-pweibull(y,shape=
    shape1,scale)^shape2
LL<-censor*log(f1(y,shape1,shape2,scale))+
    (1-censor)*log(s1(y,shape1,shape2,scale))
LL<-sum(LL)
LP<-LL+beta.prior+shape1.prior+shape2.prior
Modelout<-list(LP=LP,Dev=-2*LL,Monitor=c(LP,
          shape1,shape2),yhat=rexpweib(length(y),shape1,
          shape2,scale),parm=parm)
return(Modelout)
}
Initial.Values<-c(coef(lm(log(y)~x11+x2+x31+x41+x51+x61+x7)),
                log(1),log(1))
```

3.5.3 Fitting of the data using LaplaceApproximation

To obtain the approximated posterior summaries of exponentiated Weibull distribution, the function `LaplaceApproximation` would be used. An argument of this function is `"Method"`, the method used to get the posterior summary. Out of **19** optimization methods, we have used three, namely Nelder-Mead (1965) "NM", trust region "TR" and BFGS. Nelder-Mead is a popular algorithm and is derivative-free and is quite efficient in large-dimensional problems and trust region is also an efficient method for optimization. So, out of these two, trust region method is used in this function and its purpose is to estimate the posterior mode and negative of the Hessian matrix leads to the asymptotic varaince-covariance matrix corresponding posterior mode. The function `SIR` which uses normal distribution as a proposal with mean equal to posterior mode and variance is equal to the asymptotic variance

obtain by the inversion of the Hessin matrix. Thus SIR returns a Monte-Carlo sample from the posterior density whose summaries are reported in Table 3.2. An object with name `M1` would be assigned to the function `LaplaceApproximation`.

```
set.seed(20)
M1<-LaplaceApproximation(Model,Initial.Values,Data=MyData,
Sample=1000,Iterations=5000,Method="TR")
M1
```

3.5.4 Output of exponentiated Weibull distribution using `LaplaceApproximation`

The object `M1` gives two summaries, `Summary1` and `Summary2`. It may be noted that `Summary1` is the summary in the form of posterior mode and modal variance, which is not reported here. However, only `Summary2` is reported. As the data contain seven regressor variables and on the basis of this Bayesian analysis we will have to conclude that which regressor variable is appropriate for modelling survival data. Once we have appropriate regressor variables, Bayesian analysis of reduced form of regression model will be made. Table 3.2 provides simulated posterior summaries of each parameter. The table contains posterior mean, posterior standard deviation and 95% credible region. Table 3.2 shows that the only three out of seven regressor

Variables	Mean	SD	95% credible region
Intercept	1.744	0.465	(0.848,2.643)
Age	0.015	0.009	(-0.002, 0.0153)
Sex	0.180	0.151	(-0.116, 0.491)
Bun	-.0159	0.0018	(-0.0193, -0.012)
Ca	-0.00618	0.046	(-0.098 0.084)
Hb	0.122	0.026	(0.070, 0.174)
Pcells	-0.0008	0.002	(-0.005, 0.003)
Protein	0.588	0.144	(0.306, 0.880)
shape1	0.615	0.096	(0.454, 0.818)
shape2	3.718	1.255	(3.535, 6.652)

Table 3.2: *Posterior mean, standard errors and 95% credible region for the multiple myeloma data (n=48) by LaplaceApproximation under the assumption of exponentiated Weibull distribution.*

variables of multiple myeloma patients are significant. The covariates Bun, Hb and Protein have credible regions (-0.0193, -0.012), (0.070, 0.174) and (0.306, 0.880)

respectively, which does not include zero and hence they are appropriate regressor variable for modelling survival data. The credible region for age variable is (-0.002, 0.0153) which includes zero in it, hence is not a significant regressor variable for modelling. In the similar manner sex, Ca and Pcells also insignificant covariates. The

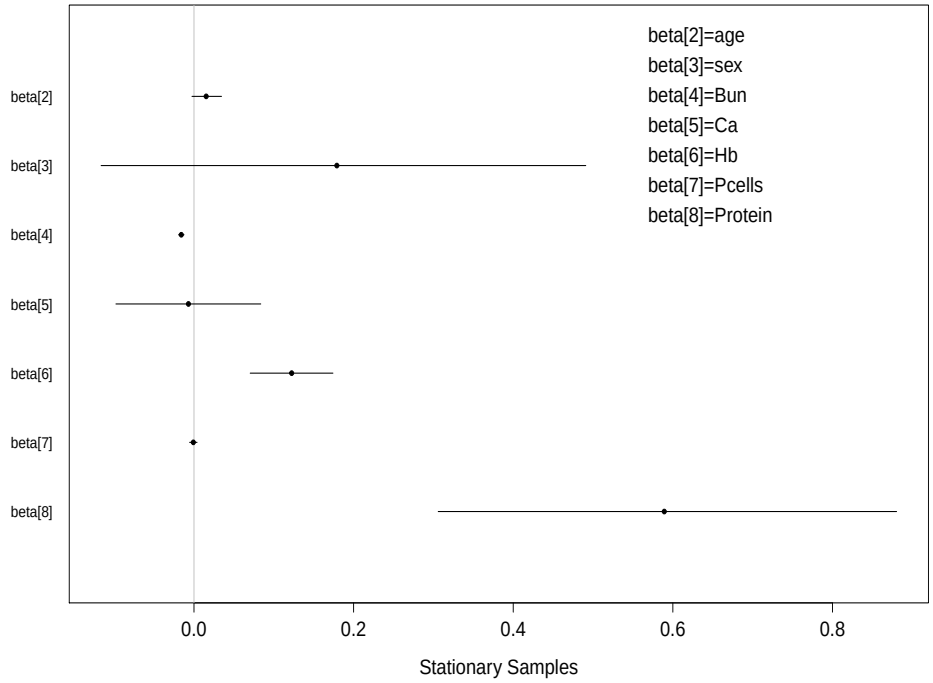

Figure 3.2: *Caterpillar plot for exponentiated Weibull distribution*

selection of appropriate regressor variable can also be done by using a caterpillar plot. Caterpillar plots are popular plots in Bayesian inference for summarizing the quantiles of posterior samples. A caterpillar plot is a horizontal plot of 3 quantiles of selected distributions. This may be used to produce a caterpillar plot of posterior samples (parameters and monitored variables). The following quantiles are plotted as a line for each parameter: 0.025 and 0.975, with the exception of a generic matrix, where unimodal 95% HPD intervals are estimated. A vertical, gray line is included at zero. The median appears as a black dot, and the quantile line is black. The starting point of the line is the 0.025 quantile and the end part of the line is 0.975 quantile. From Figure 3.2 it could be seen that -0.0193 and -0.012 is the 0.025 and 0.975 quantile of blood urea nitrogen, 0.070 and 0.174 are quantiles of Hb and 0.306 and 0.880 are also the 0.025 and 0.975 quantile of Bence-Jones protein, respectively. On the basis of these horizontal lines which represent the quantiles of the variable, we can say that only Bun (-0.0193, -0.012), Hb (0.070, 0.174) and protein (0.306, 0.880), which are not cross by the vertical line at zero.

Hence, these covariates are significant whereas rest of the variables are crossed by the vertical line which shows the insignificancy of the variables.

3.6 Bayesian analysis of exponentiated exponential distribution

Exponentiated exponential is the special case of exponentiated Weibull distribution at $\alpha = 1$. This distribution has a shape and a scale parameter. R code for the analysis of exponentiated exponential distribution are described in the following subsection.

3.6.1 R code for creation of multiple myeloma data for exponentiated exponential

Multiple myeloma survival data has been created in listed form. Vector y,x_{11},x_2,x_{31}, x_{41},x_{51},x_{61} and x_7 are the vector for survival time and regressor variables, respectively.

```
y<-c(13,52,6,40,10,7,66,10,10,14,16,4,65,5,11,10,15,5,76,56,88,24,
   51,4,40,8,18,5,16,50,40,1,36,5,10,91,18,1,18,6,1,23,15,18,12,12
   ,17,3)

censor<-c(1,0,1,1,1,0,1,0,1,1,1,1,1,1,0,1,0,1,0,0,1,1,1,1,0,1,1,1,1,
        1,1,1,1,1,1,1,0,1,0,1,1,1,1,1,0,1,1,0)
#age
x1<-c(66,66,53,69,65,57,52,60,70,70,68,50,59,60,66,51,55,67,60,66,
    63,67,60,74,72,55,51,70,53,74,70,67,63,77,61,58,69,57,59,61,75,
    56,62,60,71,60,65,59)
x11<-x1-mean(x1)
#Sex
x2<-c(0,0,1,0,0,1,0,0,0,0,0,1,0,0,1,1,0,1,0,0,0,0,1,0,0,0,0,1,0,0,1,
    0,0,0,0,1,1,0,1,1,0,1,1,1,1,1,1,0)
#Bun
x3<-c(25,13,15,10,20,12,21,41,37,40,39,172,28,13,25,12,14,26,12,18,
    21,10,10,48,57,53,12,130,17,37,14,165,40,23,13,27,21,20,21,11,
    56,20,21,18,46,6,28,90)
x31<-x3-mean(x3)
#Ca
```

```
x4<-c(10,11,13,10,10,8,10,9,12,11,10,9,9,10,9,9,9,8,12,11,9,10,10,
    9,9,12,15,8,9,13,9,10,9,8,10,11,10,9,10,10,12,9,10,9,9,10,8,
    10)
x41<-x4-mean(x4)
#Hb
x5<-c(14.6,12,11.4,10.2,13.2,9.9,12.8,14,7.5,10.6,11.2,10.1,6.6,
    9.7,8.8,9.6,13,10.4,14,12.5,14,12.4,10.1,6.5,12.8,8.2,14.4,
    10.2,10,7.7,5,9.4,11,9,14,11,10.8,5.1,13,5.1,11.3,14.6,8.8,
    7.5,4.9,5.5,7.5,10.2)
x51<-x5-mean(x5)
#Pcells
x6<-c(18,100,33,30,66,45,11,70,47,27,41,46,66,25,23,80,8,49,9,90,
    42,44,45,54,28,55,100,23,28,11,22,90,16,29,19,26,33,100,100,
    100,18,3,5,85,62,25,8,6)
x61<-x6-mean(x6)
#Protein
x7<-c(1,0,1,1,0,0,1,1,0,0,0,1,0,0,0,0,0,0,0,0,1,0,1,0,1,0,0,0,0,
    1,0,0,1,0,0,1,0,1,0,0,0,0,0,1,0,0,0,1)
library(LaplacesDemon)
X<-cbind(1,x11,x2,x31,x41,x51,x61,x7)
J<-8
mon.names<-c("LP","shape")
parm.names<-as.parm.names(list(beta=rep(0,J),log.shape=0))
MyData<-list(J=J,X=X,mon.names=mon.names,parm.names=
        parm.names,y=y,censor=censor)
```

3.6.2 R code for defining a Bayesian model for exponentiated exponential distribution.

A Bayesian model for exponentiated exponential distribution has been defined using the function `Model`. A `Model` function contians all the necessary information required to build a Bayesian model, like, defining prior for the parameters, log-likelihood and finally log posterior has been calculated as `LP`. Object `M1` is assigned for the function `LaplaceApproximation`. The output obtained by object `M1` is reported in Table 3.3 and caterpillar plot of output is provided in Figure 3.3.

```
Model<-function(parm,Data)
{
beta<-parm[1:Data$J]
```

```
shape<-exp(parm[Data$J+1])
beta.prior<-sum(dnorm(beta,0,1000,log=T))
shape.prior<-dhalfcauchy(shape,25,log=T)
mu<-tcrossprod(beta,Data$X)
scale<-exp(mu)
f1<-log(shape)-log(scale)+(shape-1)*log(1-exp(-(y/scale)))-y/scale
s1<-log(1-(1-exp(-(y/scale)))^shape)
LL<-censor*f1+(1-censor)*s1
LL<-sum(LL)
LP<-LL+beta.prior+shape.prior
Modelout<-list(LP=LP,Dev=-2*LL,Monitor=c(LP,shape),yhat=
        rexpoexp(length(y),shape,scale),parm=parm)
return(Modelout)
}
Initial.Values <-c(coef(lm(log(y)~x11+x2+x31+x41+x51+
               x61+x7)),log(1))
M1<-LaplaceApproximation(Model,Initial.Values,Data=MyData,
    Sample=10000,Iterations=10000,Method="TR")
```

Variables	Mean	SD	95% credible region
Intercept	2.851	0.104	(2.647,3.054)
Age	0.012	0.009	(-0.005,0.0304)
Sex	0.001	0.129	(-0.253,0.265)
Bun	-0.016	0.002	(-0.019,-0.012)
Ca	-0.025	0.045	(-0.113,0.064)
Hb	0.095	0.022	(0.052 , 0.139)
Pcells	0.0001	0.002	(-0.003,0.005)
Protein	0.624	0.129	(0.379,0.879)
shape	1.387	0.123	(1.162,1.651)

Table 3.3: *Posterior mean, standard errors and 95% credible region for the multiple myeloma data (n=48) by LaplaceApproximation under the assumption of exponentiated exponential distribution.*

Table 3.3 shows that the covariates Bun (-0.019, -0.012), Hb (0.052, 0.139) and Protein (0.379, 0.879) are the significant variables as they do not includes zero in their credible region and rest of the variables are not significant. Similar results were obtained in the case of exponentiated Weibull. It would be more clear by

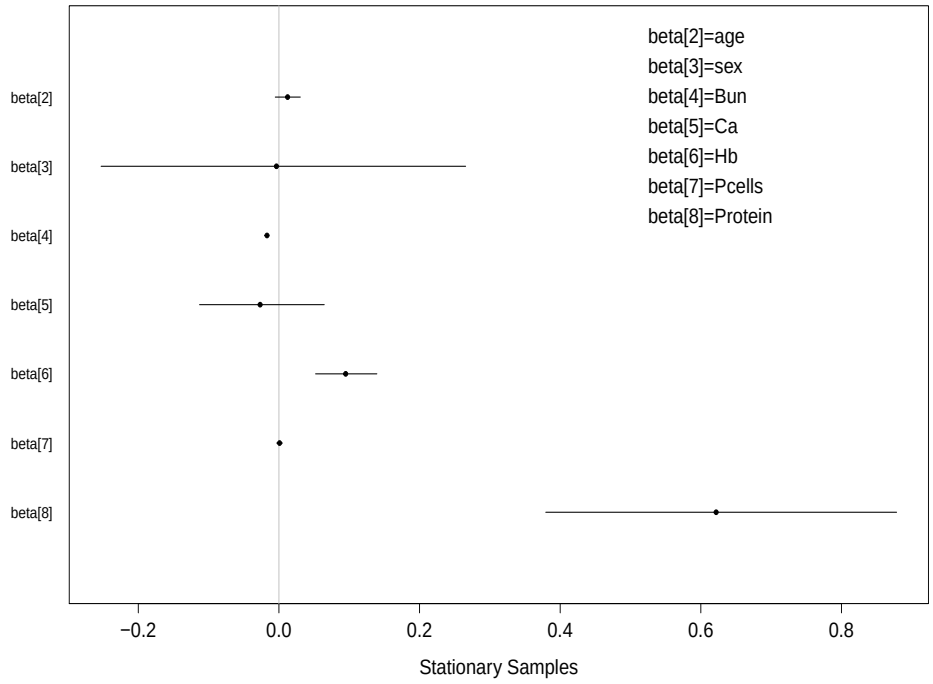

Figure 3.3: *Caterpillar plot for exponentiated exponential distribution*

graphical representation through caterpillar plot. From Figure 3.3 it could be seen that horizontal line of Bun, Hb and Protein are not cross by the vertical line at zero, whereas horizontal line of rest of the variables are crossed by the vertical line. Thus, we can say that for exponentiated exponential distriburion Bun, Hb and Protein are the significant variables, which are similar as the exponentiated Weibull model. The next model to be analysed in Bayesian framework is Burr type X distribution.

3.7 Bayesian modelling of Burr type X distribution

Burr type X is the sub model of exponentiated Weibull distribution at $\alpha = 2$. R code for the modelling of Burr type X distribution in Bayesian scenario are given as in the following steps.

3.7.1 Creation of data for the fitting of Burr type

The multiple myeloma survival data has been created with object name `MyData` for Burr type X distribution.

```
y<-c(13,52,6,40,10,7,66,10,10,14,16,4,65,5,11,10,15,5,76,56,88,24,
   51,4,40,8,18,5,16,50,40,1,36,5,10,91,18,1,18,6,1,23,15,18,12,12
   ,17,3)

censor<-c(1,0,1,1,1,0,1,0,1,1,1,1,1,1,0,1,0,1,0,0,1,1,1,1,0,1,1,1,1,
        1,1,1,1,1,1,1,0,1,0,1,1,1,1,1,0,1,1,0)
#age
x1<-c(66,66,53,69,65,57,52,60,70,70,68,50,59,60,66,51,55,67,60,66,
    63,67,60,74,72,55,51,70,53,74,70,67,63,77,61,58,69,57,59,61,75,
    56,62,60,71,60,65,59)
x11<-x1-mean(x1)
#Sex
x2<-c(0,0,1,0,0,1,0,0,0,0,0,1,0,0,1,1,0,1,0,0,0,0,1,0,0,0,0,1,0,0,1,
    0,0,0,0,1,1,0,1,1,0,1,1,1,1,1,1,0)
#Bun
x3<-c(25,13,15,10,20,12,21,41,37,40,39,172,28,13,25,12,14,26,12,18,
    21,10,10,48,57,53,12,130,17,37,14,165,40,23,13,27,21,20,21,11,
    56,20,21,18,46,6,28,90)
x31<-x3-mean(x3)
#Ca
x4<-c(10,11,13,10,10,8,10,9,12,11,10,9,9,10,9,9,9,8,12,11,9,10,10,
    9,9,12,15,8,9,13,9,10,9,8,10,11,10,9,10,10,12,9,10,9,9,10,8,
    10)
x41<-x4-mean(x4)
#Hb
x5<-c(14.6,12,11.4,10.2,13.2,9.9,12.8,14,7.5,10.6,11.2,10.1,6.6,
    9.7,8.8,9.6,13,10.4,14,12.5,14,12.4,10.1,6.5,12.8,8.2,14.4,
    10.2,10,7.7,5,9.4,11,9,14,11,10.8,5.1,13,5.1,11.3,14.6,8.8,
    7.5,4.9,5.5,7.5,10.2)
x51<-x5-mean(x5)
#Pcells
x6<-c(18,100,33,30,66,45,11,70,47,27,41,46,66,25,23,80,8,49,9,90,
    42,44,45,54,28,55,100,23,28,11,22,90,16,29,19,26,33,100,100,
    100,18,3,5,85,62,25,8,6)
x61<-x6-mean(x6)
#Protein
x7<-c(1,0,1,1,0,0,1,1,0,0,0,1,0,0,0,0,0,0,0,0,1,0,1,0,1,0,0,0,0,
    1,0,0,1,0,0,1,0,1,0,0,0,0,0,1,0,0,0,1)
X<-cbind(1,x11,x2,x31,x41,x51,x61,x7)
```

```
J<-8
mon.names<-c("LP","shape")
parm.names<-as.parm.names(list(beta=rep(0,J),log.shape=0))
MyData<-list(J=J,X=X,mon.names=mon.names,parm.names=
        parm.names,y=y,censor=censor)
```

3.7.2 Defining a Bayesian model for Burr type X distribution

In this section a Bayesian model for Burr type X distribution has been define using an object `Model`. This function has all the information regarding posterior distribution like, defining parameters of the distribution, defining f1 for density of Burr type X distribution and s1 for survival function of the same distribution to get the logposterior. The output obtained by object `M1` is reported in Table 3.4.

```
Model<-function(parm,Data)
{
beta<-parm[1:Data$J]
shape<-exp(parm[Data$J+1])
beta.prior<-sum(dnorm(beta,0,1000,log=T))
shape.prior<-dhalfcauchy(shape,25,log=T)
mu<-tcrossprod(beta,Data$X)
scale<-exp(mu)
f1<-log(2)+log(shape)-log(scale)^2+log(y)-(y/scale)^2+
(shape-1)*log(1-exp(-(y/scale)^2))
s1<-log(1-(1-exp(-y/scale)^2))
LL<-censor*f1+(1-censor)*s1
LL<-sum(LL)
LP<-LL+beta.prior+shape.prior
Modelout<-list(LP=LP,Dev=-2*LL,Monitor=c(LP,shape),
          yhat=rburr(length(y),shape,scale),parm=parm)
return(Modelout)
}
Initial.Values<-c(coef(lm(log(y)~x11+x2+x31+x41+x51+
                x61+x7)),log(1))
M1<-LaplaceApproximation(Model,Initial.Values,Data=MyData,
    Sample=10000,Iterations=10000,Method="TR")
```

Variables	Mean	SD	95% credible region
Intercept	2.832	0.081	(2.695, 3.021)
Age	0.002	0.008	(-0.012, 0.020)
Sex	-0.188	0.106	(-0.401, 0.011)
Bun	-0.013	0.002	(-0.016, -0.009)
Ca	0.012	0.036	(-0.052, 0.085)
Hb	0.042	0.016	(0.014, 0.073)
Pcells	0.0007	0.001	(-0.003, 0.004)
Protein	0.501	0.116	(0.291, 0.736)
shape	0.757	0.131	(0.532, 0.104)

Table 3.4: *Posterior mean, standard errors and 95% credible region of Burr type X distribution.*

Table 3.4 shows that the covariates Bun (0.01, 0.02), Hb (0.014, 0.073) and Protein (-1.60, -0.42) are the significant variables as they do not includes zero in their credible region. Other vriables, Age, Sex, Ca and Pcells are not significant. The statistical significance of the variable would be more clear by caterpillar plot reported in Figure 3.4. The next model to be analysed in Bayesian framework is Rayleigh distribution.

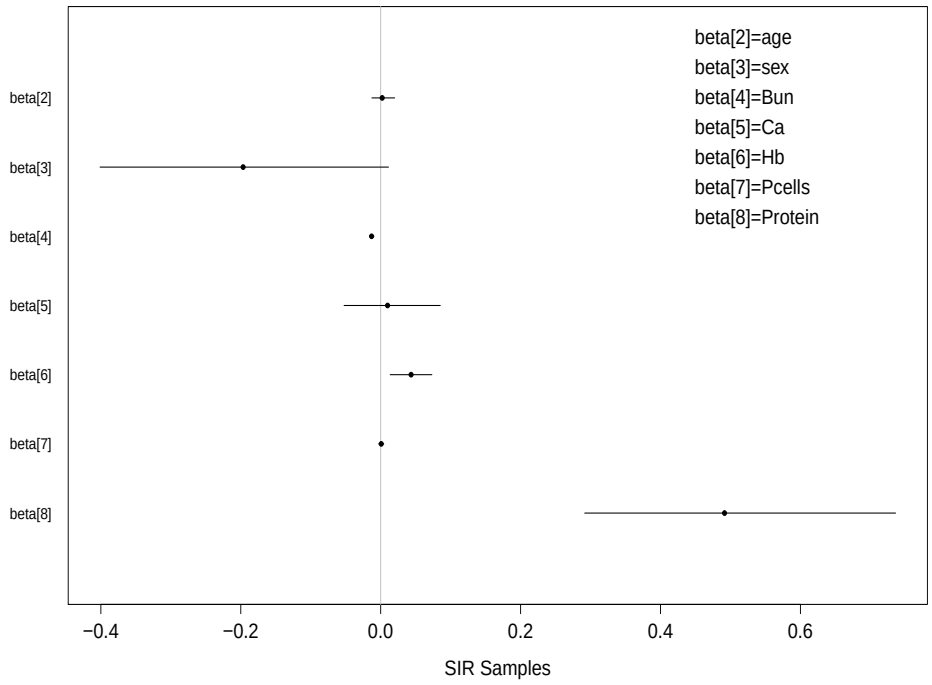

Figure 3.4: *Caterpillar plot for Burr type X distribution*

3.8 Bayesian modelling of Rayleigh distribution

Rayleigh distribution is the third sub-model of exponentiated Weibull distribution at $\alpha = 2$ and $\gamma = 1$. Its Bayesian modeling which includes R code are given as

```
y<-c(13,52,6,40,10,7,66,10,10,14,16,4,65,5,11,10,15,5,76,56,88,24,
   51,4,40,8,18,5,16,50,40,1,36,5,10,91,18,1,18,6,1,23,15,18,12,12
   ,17,3)

censor<-c(1,0,1,1,1,0,1,0,1,1,1,1,1,1,0,1,0,1,0,0,1,1,1,1,0,1,1,1,1,
        1,1,1,1,1,1,1,0,1,0,1,1,1,1,1,0,1,1,0)
#age
x1<-c(66,66,53,69,65,57,52,60,70,70,68,50,59,60,66,51,55,67,60,66,
    63,67,60,74,72,55,51,70,53,74,70,67,63,77,61,58,69,57,59,61,75,
    56,62,60,71,60,65,59)
x11<-x1-mean(x1)
#Sex
x2<-c(0,0,1,0,0,1,0,0,0,0,0,1,0,0,1,1,0,1,0,0,0,0,1,0,0,0,0,1,0,0,1,
    0,0,0,0,1,1,0,1,1,0,1,1,1,1,1,1,0)
#Bun
x3<-c(25,13,15,10,20,12,21,41,37,40,39,172,28,13,25,12,14,26,12,18,
    21,10,10,48,57,53,12,130,17,37,14,165,40,23,13,27,21,20,21,11,
    56,20,21,18,46,6,28,90)
x31<-x3-mean(x3)
#Ca
x4<-c(10,11,13,10,10,8,10,9,12,11,10,9,9,10,9,9,9,8,12,11,9,10,10,
    9,9,12,15,8,9,13,9,10,9,8,10,11,10,9,10,10,12,9,10,9,9,10,8,
    10)
x41<-x4-mean(x4)
#Hb
x5<-c(14.6,12,11.4,10.2,13.2,9.9,12.8,14,7.5,10.6,11.2,10.1,6.6,
    9.7,8.8,9.6,13,10.4,14,12.5,14,12.4,10.1,6.5,12.8,8.2,14.4,
    10.2,10,7.7,5,9.4,11,9,14,11,10.8,5.1,13,5.1,11.3,14.6,8.8,
    7.5,4.9,5.5,7.5,10.2)
x51<-x5-mean(x5)
#Pcells
x6<-c(18,100,33,30,66,45,11,70,47,27,41,46,66,25,23,80,8,49,9,90,
    42,44,45,54,28,55,100,23,28,11,22,90,16,29,19,26,33,100,100,
    100,18,3,5,85,62,25,8,6)
```

```
x61<-x6-mean(x6)
#Protein
x7<-c(1,0,1,1,0,0,1,1,0,0,0,1,0,0,0,0,0,0,0,0,1,0,1,0,1,0,0,0,0,
    1,0,0,1,0,0,1,0,1,0,0,0,0,0,1,0,0,0,1)
X<-cbind(1,x11,x2,x31,x41,x51,x61,x7)
J<-8
mon.names<-c("LP")
parm.names<-as.parm.names(list(beta=rep(0,J)))
MyData<-list(J=J,X=X,mon.names=mon.names,parm.names=
        parm.names,y=y,censor=censor)
Initial.Values<-c(coef(lm(log(y)~x11+x2+x31+x41+x51+
                x61+x7)))
Model<-function(parm,Data)
{
beta<-parm[1:Data$J]
beta.prior<-sum(dnorm(beta,0,1000,log=T))
mu<-tcrossprod(beta,Data$X)
f1<-log(2)-log(scale)^2+log(y)-(y/scale)^2
s1<--(y/scale)^2
LL<-censor*f1+(1-censor)*s1
LL<-sum(LL)
LP<-LL+beta.prior
Modelout<-list(LP=LP,Dev=-2*LL,Monitor=c(LP),
          yhat=rray(length(y),scale),parm=parm)
return(Modelout)
}
M1<-LaplaceApproximation(Model,Initial.Values,Data=MyData,
    Sample=10000,Iterations=10000,Method="TR")
```

Here also on the basis of Bayesian analysis of Rayleigh distribution summarized in Table 3.5 and through caterpillar plot reported in Figure 3.5, we found Bun, Hb and Protein are appropriate regressors to be included in the model.

3.9 Bayesian modelling of Weibull distribution

Weibull distribution is the fourth and important sub-model of exponentiated Weibull distribution at $\alpha = 1$. The R code for the modelling of Weibull distribution are

Variables	Mean	SD	95% credible region
Intercept	2.957	0.151	(2.66, 3.26)
Age	0.002	0.01	(-0.013, 0.020)
Sex	-0.183	0.110	(-0.390, 0.030)
Bun	-0.014	0.001	(-0.021, -0.012)
Ca	0.012	0.046	(-0.071, 0.084)
Hb	0.036	0.017	(0.001, 0.083)
Pcells	0.002	0.001	(-0.001, 0.002)
Protein	0.512	0.118	(0.301, 0.722)

Table 3.5: *Posterior mean, standard errors and 95% credible region of Rayleigh distribution.*

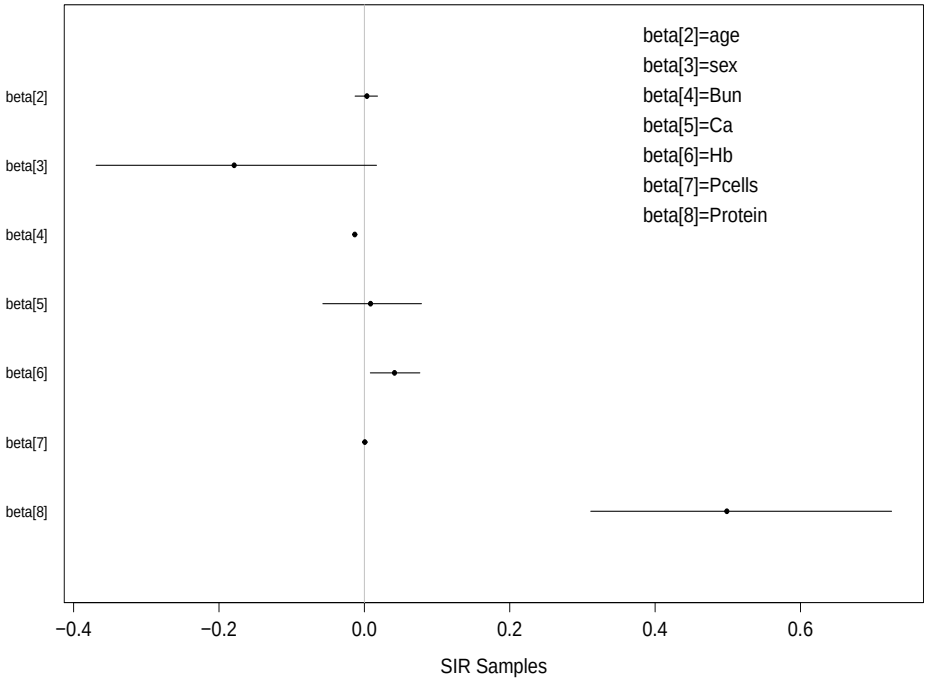

Figure 3.5: *Caterpillar plot for Rayleigh distribution*

```
y<-c(13,52,6,40,10,7,66,10,10,14,16,4,65,5,11,10,15,5,76,56,88,24,
   51,4,40,8,18,5,16,50,40,1,36,5,10,91,18,1,18,6,1,23,15,18,12,12
   ,17,3)

censor<-c(1,0,1,1,1,0,1,0,1,1,1,1,1,1,0,1,0,1,0,0,1,1,1,1,0,1,1,1,1,
        1,1,1,1,1,1,1,0,1,0,1,1,1,1,1,0,1,1,0)
#age
x1<-c(66,66,53,69,65,57,52,60,70,70,68,50,59,60,66,51,55,67,60,66,
```

```
    63,67,60,74,72,55,51,70,53,74,70,67,63,77,61,58,69,57,59,61,75,
    56,62,60,71,60,65,59)
x11<-x1-mean(x1)
#Sex
x2<-c(0,0,1,0,0,1,0,0,0,0,0,1,0,0,1,1,0,1,0,0,0,0,1,0,0,0,0,1,0,0,1,
    0,0,0,0,1,1,0,1,1,0,1,1,1,1,1,1,0)
#Bun
x3<-c(25,13,15,10,20,12,21,41,37,40,39,172,28,13,25,12,14,26,12,18,
    21,10,10,48,57,53,12,130,17,37,14,165,40,23,13,27,21,20,21,11,
    56,20,21,18,46,6,28,90)
x31<-x3-mean(x3)
#Ca
x4<-c(10,11,13,10,10,8,10,9,12,11,10,9,9,10,9,9,9,8,12,11,9,10,10,
    9,9,12,15,8,9,13,9,10,9,8,10,11,10,9,10,10,12,9,10,9,9,10,8,
    10)
x41<-x4-mean(x4)
#Hb
x5<-c(14.6,12,11.4,10.2,13.2,9.9,12.8,14,7.5,10.6,11.2,10.1,6.6,
    9.7,8.8,9.6,13,10.4,14,12.5,14,12.4,10.1,6.5,12.8,8.2,14.4,
    10.2,10,7.7,5,9.4,11,9,14,11,10.8,5.1,13,5.1,11.3,14.6,8.8,
    7.5,4.9,5.5,7.5,10.2)
x51<-x5-mean(x5)
#Pcells
x6<-c(18,100,33,30,66,45,11,70,47,27,41,46,66,25,23,80,8,49,9,90,
    42,44,45,54,28,55,100,23,28,11,22,90,16,29,19,26,33,100,100,
    100,18,3,5,85,62,25,8,6)
x61<-x6-mean(x6)
#Protein
x7<-c(1,0,1,1,0,0,1,1,0,0,0,1,0,0,0,0,0,0,0,0,1,0,1,0,1,0,0,0,0,
    1,0,0,1,0,0,1,0,1,0,0,0,0,0,1,0,0,0,1)
X<-cbind(1,x11,x2,x31,x41,x51,x61,x7)
J<-8
mon.names<-c("LP","shape")
parm.names<-as.parm.names(list(beta=rep(0,J),
            log.shape=0))
MyData<-list(J=J,X=X,mon.names=mon.names,parm.names=
        parm.names,y=y,censor=censor)
Initial.Values<-c(coef(lm(log(y)~x11+x2+x31+x41+x51+x61+x7)),
                log(1))
```

```
Model<-function(parm,Data)
{
#Parameters
beta<-parm[1:Data$J]
shape<-exp(parm[Data$J+1])
# Log(Prior Densities)
beta.prior<-sum(dnorm(beta,0,sqrt(1000),log=T))
shape.prior<-dhalfcauchy(shape,20,log=T)
# Loglikelihood
mu<-tcrossprod(beta,Data$X)
scale<-exp(mu)
LL<-censor*dweibull(Data$y,shape,scale,log=T)+
(1-censor)*pweibull(Data$y,shape,scale,log.p=T,lower.tail=F)
LL<-sum(LL)
## Log-posterior
LP<-LL+beta.prior+shape.prior
Modelout<-list(LP=LP,Dev=-2*LL,Monitor=c(LP,shape),
          yhat=rweibull(length(y),shape,scale),parm=parm)
return(Modelout)
}
M1<-LaplaceApproximation(Model,Initial.Values,Data=
    MyData,Method="TR",Iterations=10000)
```

Variables	Mean	SD	95% credible region
Intercept	3.074	0.142	(2.807, 3.350)
Age	0.013	0.009	(-0.005, 0.031)
Sex	-0.028	0.135	(-0.291, 0.234)
Bun	-0.017	0.002	(-0.020, -0.014)
Ca	-0.022	0.045	(-0.107, 0.073)
Hb	0.092	0.024	(0.046, 0.139)
Pcells	0.001	0.002	(-0.003,0.005)
Protein	0.614	0.137	(0.348,0.873)
shape	1.168	0.065	(1.044,1.302)

Table 3.6: *Posterior mean, standard errors and 95% credible region for the multiple myeloma data (n=48) under the assumption of Weibull distribution.*

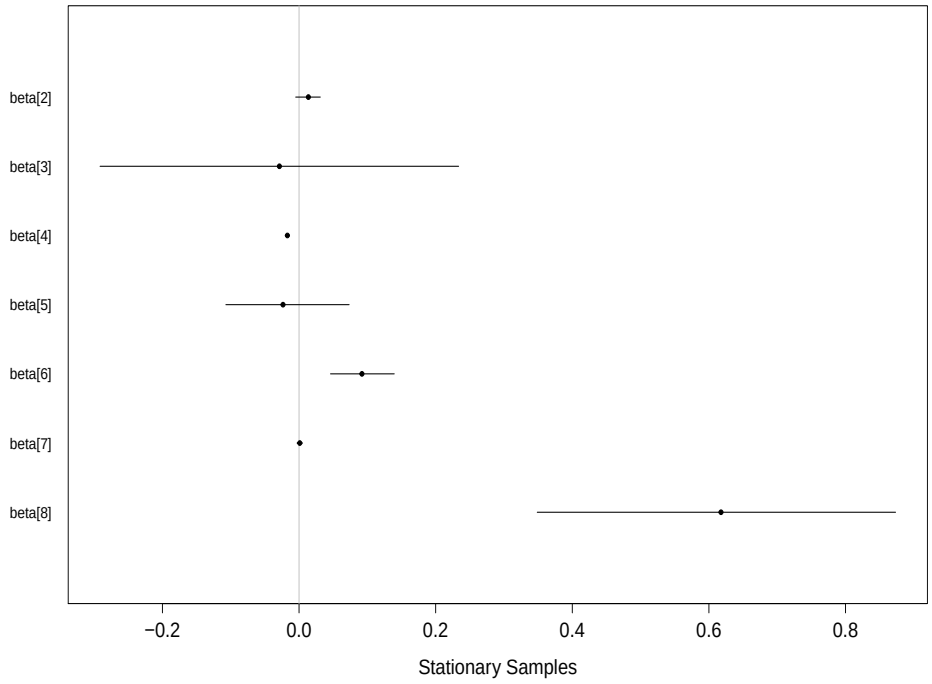

Figure 3.6: *Caterpillar plot for Weibull distribution*

After Bayesian analysis of exponentiated Weibull, exponentiated exponential, Burr type X, Rayleigh and Weibull distribution, we are successfully able to figure out that which regressor variable is significant and appropriate to be in the model. On the basis of Bayesian analysis of EW and its sub models, Table 3.2, 3.3, 3.4, 3.5 and 3.6 and through caterpillar plots of respective distributions, shows that the variables `Bun`, `Hb` and `Protein` are significant and play an important role in the model. Hence, it would be concluded that `Bun`, `Hb` and `Protein` are important regressors and can not be excluded from the model. So, now we will do the Bayesian analysis of reduced form of regressor model by approximation and simulation tools using `LaplacesDemon` package.

3.10 Bayesian analysis of reduced form of exponentiated Weibull distribution

On the basis of previous Bayesian analysis, only three out of seven regressor variables are suitable for regression modelling. Now, in this section we have analyzed a reduced form of regression model containing three covariates with censoring mechanism for exponentiated Weibull distribution. The analysis includes `R` code for creating data, definition of Bayesian model and fitting of data with `LaplaceApproximation` and `LaplacesDemon` functions. `M1` is the object assigned by making use of `LaplaceApproximation` and object `M2` is assigned by making use of `LaplacesDemon`. The output obtained by `LaplaceApproximation` is reported in Table 3.7 and 3.8 and simulated posterior obtained by `LaplacesDemon` is given in Table 3.9. Table 3.8 gives the simulated posterior summary by sampling importance resampling using function `LaplaceApproximation` and Table 3.9 is also the simulated posterior summary obtained by independent Metropolis algorithm. For reduced model `R` code are described as

```
y<-c(13,52,6,40,10,7,66,10,10,14,16,4,65,5,11,10,15,5,76,56,88,24,
   51,4,40,8,18,5,16,50,40,1,36,5,10,91,18,1,18,6,1,23,15,18,12,
   12,17,3)

censor<-c(1,0,1,1,1,0,1,0,1,1,1,1,1,1,0,1,0,1,0,0,1,1,1,1,0,1,1,1,
        1,1,1,1,1,1,1,1,0,1,0,1,1,1,1,1,0,1,1,0)

#####Blood urea nitrogen(Bun)#####
x1<-c(25,13,15,10,20,12,21,41,37,40,39,172,28,13,25,12,14,26,12,18,
    21,10,10,48,57,53,12,130,17,37,14,165,40,23,13,27,21,20,21,11,
    56,20,21,18,46,6,28,90)
x11<-x1-mean(x1)

##### Haemoglobin(Hb)####
x2<-c(14.6,12,11.4,10.2,13.2,9.9,12.8,14,7.5,10.6,11.2,10.1,6.6,
    9.7,8.8,9.6,13,10.4,14,12.5,14,12.4,10.1,6.5,12.8,8.2,14.4,
    10.2,10,7.7,5,9.4,11,9,14,11,10.8,5.1,13,5.1,11.3,14.6,8.8,
    7.5,4.9,5.5,7.5,10.2)
x21<-x2-mean(x2)

#### Bence-Jones protein####
```

```
x3<-c(1,0,1,1,0,0,1,1,0,0,0,1,0,0,0,0,0,0,0,0,1,0,1,0,1,0,0,0,0,
    1,0,0,1,0,0,1,0,1,0,0,0,0,0,1,0,0,0,1)
X<-cbind(1,x11,x21,x3)
J<-4
mon.names<-c("LP","shape1","shape2")
parm.names<-as.parm.names(list(beta=rep(0,J),
            log.shape1=0,log.shape2=0))
MyData<-list(J=J,X=X,mon.names=mon.names,parm.names=
        parm.names,y=y,censor=censor)
Model<-function(parm,Data)
{
beta<-parm[1:Data$J]
shape1<-exp(parm[Data$J+1])
shape2<-exp(parm[Data$J+2])
beta.prior<-sum(dnorm(beta,0,1000,log=T))
shape1.prior<-dhalfcauchy(shape1,25,log=T)
shape2.prior<-dhalfcauchy(shape2,25,log=T)
mu<-tcrossprod(beta,Data$X)
scale<-exp(mu)
dy<-function(y,shape1,shape2,scale) shape2*dweibull(y,shape=shape1,
    scale)*pweibull(y,shape=shape1,scale)^(shape2-1)
sy<-function(y,shape1,shape2,scale) 1-pweibull(y,
    shape=shape1,scale)^shape2
LL<-censor*log(dy(y,shape1,shape2,scale))+
    (1-censor)*log(sy(y,shape1,shape2,scale))
LL<-sum(LL)
LP<-LL+beta.prior+shape1.prior+shape2.prior
Modelout<-list(LP=LP,Dev=-2*LL,Monitor=c(LP,shape1,
          shape2),yhat=rexpweib(length(y),
          shape1,shape2,scale),parm=parm)
return(Modelout)
}
Initial.Values<-c(coef(lm(log(y)~x11+x21+x3)),log(1),log(1))
set.seed(300)
M1<-LaplaceApproximation(Model,Initial.Values,Data=MyData,
    Sample=1000,Iterations=5000,Method="TR")

Initial.Values<-as.initial.values(M1)
M2<-LaplacesDemon(Model, Data=MyData, Initial.Values,
```

	Mode	SD	LB	UB
Intercept	2.19	1.24	-0.29	4.67
Bun	-0.02	0.00	-0.02	-0.01
Hb	0.10	0.06	-0.02	0.21
Protein	0.60	0.33	-0.06	1.27
log.shape1	-0.35	0.53	-1.41	0.72
log.shape2	0.97	1.06	-1.15	3.10

Table 3.7: *Aprroximated posterior summary of reduced form of regression model of exponentiated Weibull distribution.*

```
Covar=M1$Covar, Iterations=2000, Status=100, Thinning=1,
Algorithm="IM",Specs=list(mu=M1$Summary1[1:length(Initial.Values),1]))
```

Table 3.7 and Table 3.8 are the output obtain by using function `LaplaceApproximation`. Table 3.7 consists of four columns, which are posterior mode, posterior sd and respective quantiles of the distribution for multiple myeloma data. Table 3.8 consists of seven columns which are posterior mean as well as posterior median of the distribution. MSCE is the Monte Carlo standard error which measures the variability of each estimate due to the simulation. The small value of MSCE represents the convergence of algorithm and ESS is the effective sample size which gives an estimate of the equivalent number of independent iterations that the chain represents. Table 3.9 refers to analysis obtained by `LaplacesDemon`. It includes both monitoring procedure of the algorithm's convergence and analysis of the sample used for the description of the posterior distribution and inference about the parameters of interest. `LaplacesDemon` function gives three plots for each

	Mean	SD	MCSE	ESS	LB	Median	UB
Intercept	1.99	0.81	0.03	1000.00	0.59	2.02	3.45
Bun	-0.02	0.00	0.00	1000.00	-0.03	-0.02	-0.01
Hb	0.10	0.06	0.00	1000.00	-0.02	0.09	0.21
Protein	0.69	0.50	0.02	1000.00	-0.21	0.64	1.95
shape1	0.67	0.23	0.01	1000.00	0.42	0.59	1.22
shape2	3.25	1.77	0.06	1000.00	0.85	3.07	6.67

Table 3.8: *Simulated posterior summary of reduced form of regression model by sampling importance resampling using the same function.*

variables that is, trace plot, density plot and auto-correlation plot which could be seen in Figure 3.7 and 3.8. Each row corresponds to one variable. In Figure 3.7,

	Mean	SD	MCSE	ESS	LB	Median	UB
Intercept	2.06	0.37	0.01	1000.00	1.33	2.07	2.76
Bun	-0.02	0.00	0.00	891.56	-0.02	-0.02	-0.01
Hb	0.10	0.02	0.00	1000.00	0.05	0.10	0.14
Protein	0.62	0.13	0.00	1000.00	0.36	0.61	0.90
shape1	0.68	0.10	0.00	1000.00	0.50	0.67	0.90
shape2	3.00	0.96	0.03	1000.00	1.58	2.86	5.38

Table 3.9: *Simulated posterior summary of reduced form of regression model by independent Metropolis algorithm using LaplacesDemon function.*

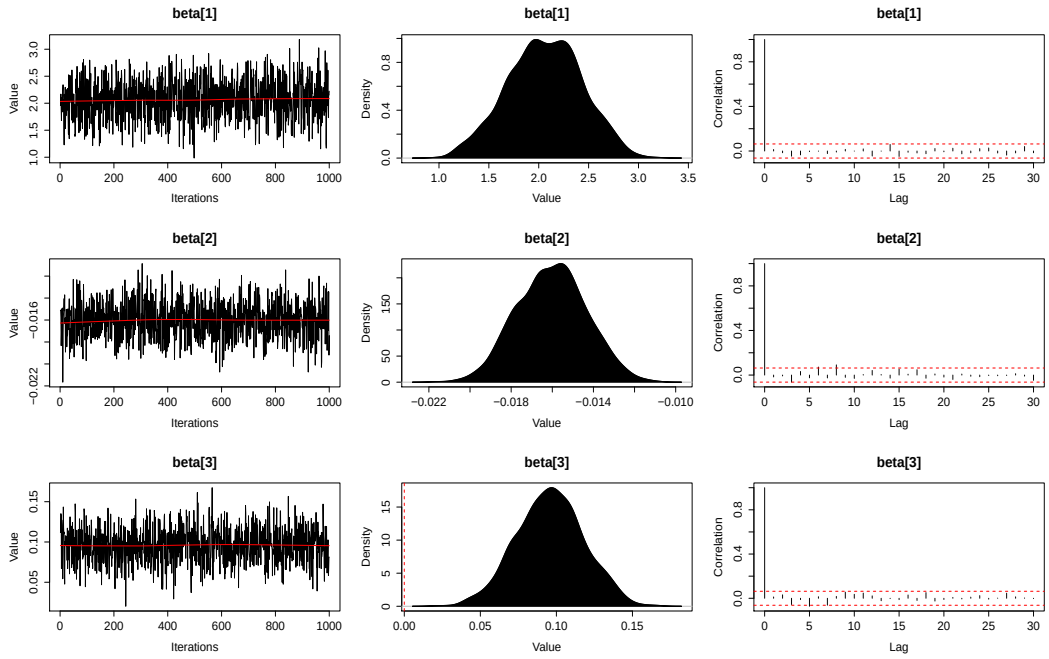

Figure 3.7: *Simulated posterior density plots with trace and auto corrrelation plots of regressor variables obtained by independent Metropolis algorithm.*

leftmost represents the trace plot which shows the values the parameter took during the runtime of the chain and the rightmost is the auto-correlation plot. Posterior density plots of `Bun`, `Hb` and `Protein` could be seen at middle. Basically, it is the (smoothened) histogram of the values in the trace-plot, i.e. the distribution of the values of the parameter in the chain. The convergence of algorithm can be seen through visual inspection of trace plot. Trace plots of `Bun`, `Hb` and `Protein` are quite convincing in terms of covergence of algorithm as the chains are mix well and they are in parallel zone. Hence, it could be concluded that the independent Metropolis algorithm is very effective with acceptance probability 36%.

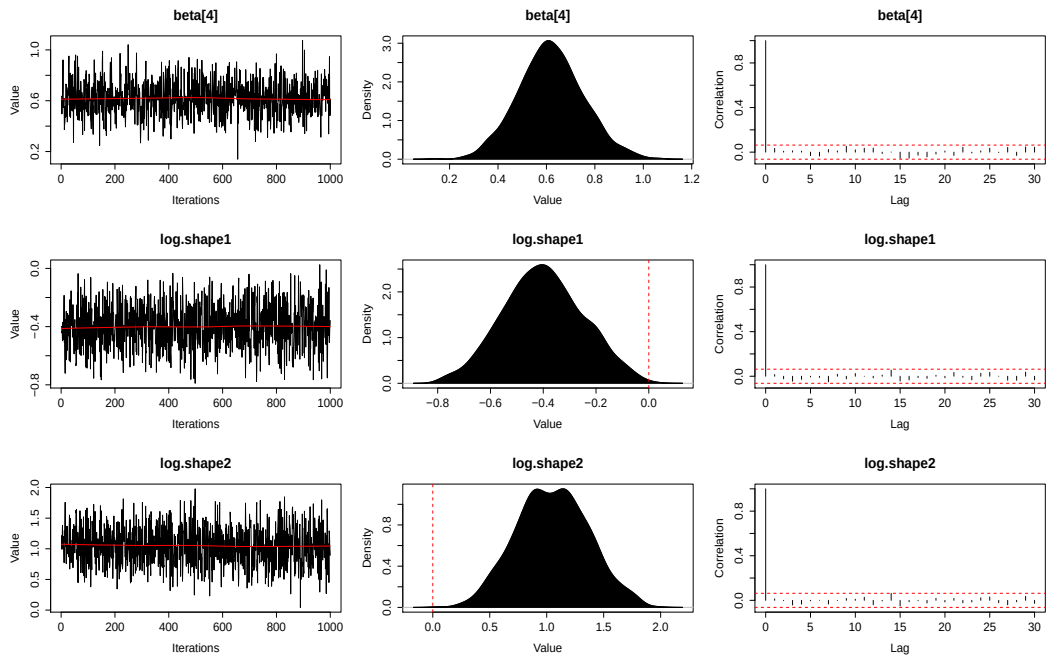

Figure 3.8: *Simulated posterior density plots with trace and auto corrrelation plots of regressor variables obtained by independent Metropolis algorithm.*

3.11 Bayesian modelling of reduced form of regression model for exponentiated exponential distribution

Bayesian analysis of exponentiated exponential distribution in reduced form of regression model is being describe in following `R` code. The approximated posterior summary is listed in Table 3.10 by trust region method using `LaplaceApproximation`. Table 3.11 is obtained by sampling importance resampling using the same function and Table 3.12 is the simulated posterior obtained by independent Metropolis algorithm using `LaplacesDemon` function.

```
y<-c(13,52,6,40,10,7,66,10,10,14,16,4,65,5,11,10,15,5,76,56,88,24,
   51,4,40,8,18,5,16,50,40,1,36,5,10,91,18,1,18,6,1,23,15,18,12,
   12,17,3)

censor<-c(1,0,1,1,1,0,1,0,1,1,1,1,1,1,0,1,0,1,0,0,1,1,1,1,0,1,1,1,
        1,1,1,1,1,1,1,1,0,1,0,1,1,1,1,1,0,1,1,0)

#####Blood urea nitrogen(Bun)#####
x1<-c(25,13,15,10,20,12,21,41,37,40,39,172,28,13,25,12,14,26,12,18,
```

```
    21,10,10,48,57,53,12,130,17,37,14,165,40,23,13,27,21,20,21,11,
    56,20,21,18,46,6,28,90)
x11<-x1-mean(x1)

##### Haemoglobin(Hb)####
x2<-c(14.6,12,11.4,10.2,13.2,9.9,12.8,14,7.5,10.6,11.2,10.1,6.6,
    9.7,8.8,9.6,13,10.4,14,12.5,14,12.4,10.1,6.5,12.8,8.2,14.4,
    10.2,10,7.7,5,9.4,11,9,14,11,10.8,5.1,13,5.1,11.3,14.6,8.8,
    7.5,4.9,5.5,7.5,10.2)
x21<-x2-mean(x2)

#### Bence-Jones protein####
x3<-c(1,0,1,1,0,0,1,1,0,0,0,1,0,0,0,0,0,0,0,0,1,0,1,0,1,0,0,0,0,
    1,0,0,1,0,0,1,0,1,0,0,0,0,0,1,0,0,0,1)
X<-cbind(1,x11,x21,x3)
J<-4
mon.names<-c("LP","shape")
parm.names<-as.parm.names(list(beta=rep(0,J),log.shape=0))
MyData<-list(J=J,X=X,mon.names=mon.names,parm.names=parm.names,
        y=y,censor=censor)
Model<-function(parm,Data)
{
  beta<-parm[1:Data$J]
  shape<-exp(parm[Data$J+1])
  beta.prior<-sum(dnorm(beta,0,1000,log=T))
  shape.prior<-dhalfcauchy(shape,25,log=T)
  mu<-tcrossprod(beta,Data$X)
  scale<-exp(mu)
  f1=log(shape)-log(scale)+(shape-1)*log(1-exp(-(y/scale)))-y/scale
  s1<-log(1-(1-exp(-(y/scale)))^shape)
  LL<-censor*f1+(1-censor)*s1
  LL<-sum(LL)
  LP<-LL+beta.prior+shape.prior
  Modelout<-list(LP=LP,Dev=-2*LL,Monitor=c(LP,shape),yhat=rexpoexp
            (length(y),shape,scale),parm=parm)
  return(Modelout)
}
Initial.Values<-c(coef(lm(log(y)~x11+x21+x3)),log(1))
```

```
M1<-LaplaceApproximation(Model,Initial.Values,Data=MyData,
    Sample=10000,Iterations=10000,Method="TR")
M1
Initial.Values<-as.initial.values(M2)
M2<-LaplacesDemon(Model,Data=MyData,Initial.Values,
Covar=M2$Covar,Iterations=22000,Status=148,Thinning=11,
Algorithm="IM",Specs=list(mu=apply(M2$Posterior1,2,mean)))
```

	Mode	SD	LB	UB
Intercept	2.83	0.24	2.34	3.32
Bun	-0.02	0.00	-0.02	-0.01
Hb	0.09	0.05	-0.02	0.19
Protein	0.60	0.32	-0.03	1.23
log.shape	0.34	0.22	-0.10	0.78

Table 3.10: *Aprroximated posterior summary of reduced form of regression model of exponentiated exponential distribution.*

	Mean	SD	MCSE	ESS	LB	Median	UB
Intercept	2.96	0.26	0.00	10000.00	2.49	2.95	3.51
Bun	-0.02	0.00	0.00	10000.00	-0.02	-0.02	-0.01
Hb	0.09	0.05	0.00	10000.00	-0.01	0.09	0.20
Protein	0.60	0.34	0.00	10000.00	-0.08	0.61	1.25
shape	1.30	0.29	0.00	10000.00	0.83	1.27	1.97

Table 3.11: *Simulated posterior summary of reduced form of regression model by sampling importance resampling using the same function.*

	Mean	SD	MCSE	ESS	LB	Median	UB
Intercept	2.85	0.10	0.00	1770.07	2.67	2.85	3.04
Bun	-0.02	0.00	0.00	1698.81	-0.02	-0.02	-0.01
Hb]	0.09	0.02	0.00	2000.00	0.05	0.09	0.13
Protein	0.60	0.12	0.00	1837.72	0.36	0.60	0.84
shape	1.38	0.12	0.00	1837.09	1.15	1.38	1.64

Table 3.12: *Simulated posterior summary of reduced form of regression model by independent Metropolis algorithm using LaplacesDemon function.*

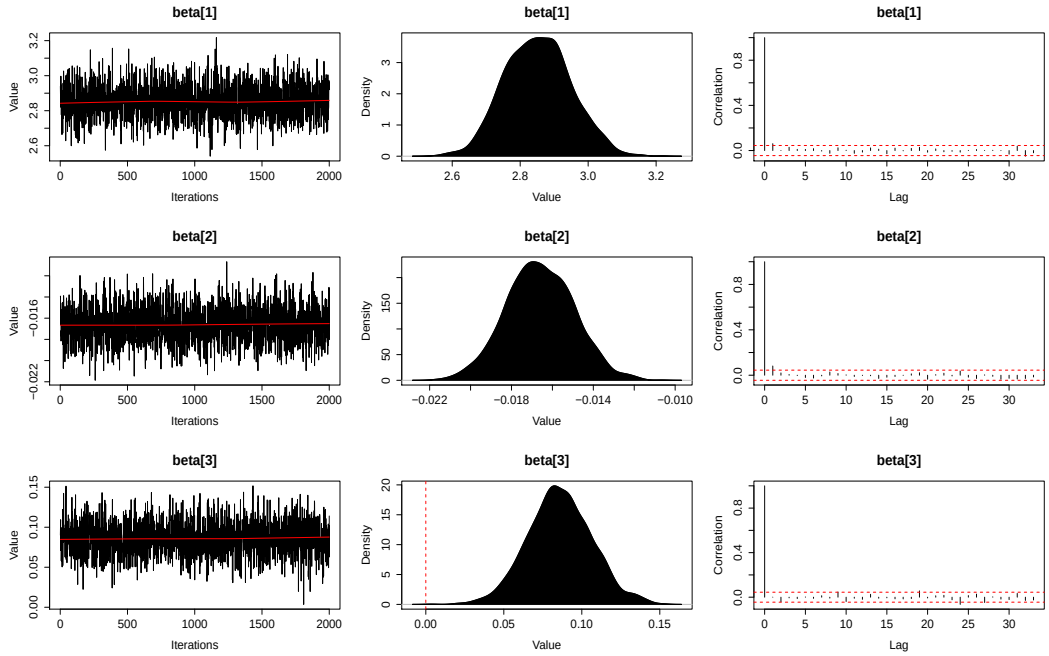

Figure 3.9: *Trace, posterior density and auto-correlation plots for variables of exponentiated exponential distribution by independent Metropolis algorithm.*

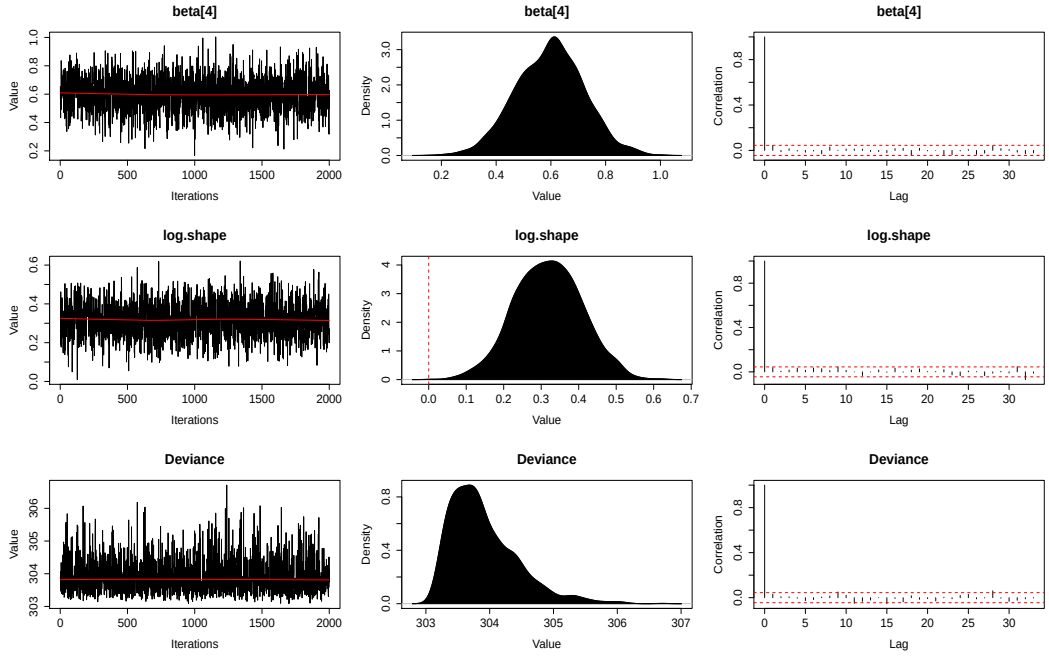

Figure 3.10: *Trace, posterior density and auto-correlation plots for variables of exponentiated exponential distribution by independent Metropolis algorithm.*

3.12 Bayesian Modelling of reduced form of regression model for Burr Type X distrubtion

Burr type X distribution is implemented in `LaplacesDemon` package for multiple myeloma data which contains censoring mechanism. Approximated and simulated posterior densities, which contain posterior mode, posterior mean, sd and respective quantiles, is reported in Table 3.13, 3.14 and 3.15 respectively.

```
y<-c(13,52,6,40,10,7,66,10,10,14,16,4,65,5,11,10,15,5,76,56,88,24,
   51,4,40,8,18,5,16,50,40,1,36,5,10,91,18,1,18,6,1,23,15,18,12,
   12,17,3)

censor<-c(1,0,1,1,1,0,1,0,1,1,1,1,1,1,0,1,0,1,0,0,1,1,1,1,0,1,1,1,
        1,1,1,1,1,1,1,1,0,1,0,1,1,1,1,1,0,1,1,0)

#####Blood urea nitrogen(Bun)#####
x1<-c(25,13,15,10,20,12,21,41,37,40,39,172,28,13,25,12,14,26,12,18,
    21,10,10,48,57,53,12,130,17,37,14,165,40,23,13,27,21,20,21,11,
    56,20,21,18,46,6,28,90)
x11<-x1-mean(x1)

##### Haemoglobin(Hb)####
x2<-c(14.6,12,11.4,10.2,13.2,9.9,12.8,14,7.5,10.6,11.2,10.1,6.6,
    9.7,8.8,9.6,13,10.4,14,12.5,14,12.4,10.1,6.5,12.8,8.2,14.4,
    10.2,10,7.7,5,9.4,11,9,14,11,10.8,5.1,13,5.1,11.3,14.6,8.8,
    7.5,4.9,5.5,7.5,10.2)
x21<-x2-mean(x2)

#### Bence-Jones protein####
x3<-c(1,0,1,1,0,0,1,1,0,0,0,1,0,0,0,0,0,0,0,0,1,0,1,0,1,0,0,0,0,
    1,0,0,1,0,0,1,0,1,0,0,0,0,0,1,0,0,0,1)

X<-cbind(1,x11,x21,x3)
J<-4
mon.names<-c("LP","shape")
parm.names<-as.parm.names(list(beta=rep(0,J),log.shape=0))
MyData<-list(J=J,X=X,mon.names=mon.names,parm.names=parm.names,
```

```
        y=y,censor=censor)

Model<-function(parm,Data)
{
beta<-parm[1:Data$J]
shape<-exp(parm[Data$J+1])
beta.prior<-sum(dnorm(beta,0,1000,log=T))
shape.prior<-dhalfcauchy(shape,25,log=T)
mu<-tcrossprod(beta,Data$X)
scale<-exp(mu)
f1<-log(2)+log(shape)-log(scale)^2+log(y)-(y/scale)^2+
(shape-1)*log(1-exp(-(y/scale)^2))
s1<-log(1-(1-exp(-y/scale)^2))
LL<-censor*f1+(1-censor)*s1
LL<-sum(LL)
LP<-LL+beta.prior+shape.prior
Modelout<-list(LP=LP,Dev=-2*LL,Monitor=c(LP,shape),yhat=rburr(
          length(y),shape,scale),parm=parm)
return(Modelout)
}
Initial.Values<-c(coef(lm(log(y)~x11+x21+x3)),log(1))
set.seed(20)
M1<-LaplaceApproximation(Model,Initial.Values,Data=MyData,
    Sample=10000,Iterations=10000,Method="TR")
M1

Initial.Values<-as.initial.values(M1)
M2<-LaplacesDemon(Model, Data=MyData, Initial.Values,
Covar=M1$Covar,Iterations=2000,Status=100,Thinning=1,
Algorithm="IM",Specs=list(mu=M1$Summary1[1:length(Initial.Values),1]))
```

	Mode	SD	LB	UB
Intercept	2.59	0.07	2.45	2.74
Bun	-0.01	0.00	-0.02	-0.01
Hb	0.01	0.02	-0.03	0.05
Protein	0.62	0.11	0.40	0.84
log.shape	-0.07	0.17	-0.41	0.27

Table 3.13: *Aprroximated posterior summary of reduced form of regression model of Burr type X distribution.*

	Mean	SD	MCSE	ESS	LB	Median	UB
Intercept	2.60	0.07	0.00	10000.00	2.46	2.60	2.75
Bun	-0.01	0.00	0.00	10000.00	-0.02	-0.01	-0.01
Hb	0.01	0.02	0.00	10000.00	-0.03	0.01	0.04
Protein	0.64	0.11	0.00	10000.00	0.43	0.65	0.87
shape	0.86	0.15	0.00	10000.00	0.58	0.85	1.19

Table 3.14: *Simulated posterior summary of reduced form of regression model by sampling importance resampling using the same function.*

	Mean	SD	MCSE	ESS	LB	Median	UB
Intercept	2.59	0.03	0.00	2000.00	2.53	2.59	2.64
Bun	-0.01	0.00	0.00	2000.00	-0.01	-0.01	-0.01
Hb	0.01	0.01	0.00	1712.51	-0.01	0.01	0.02
Protein	0.63	0.04	0.00	2000.00	0.55	0.63	0.72
shape	0.91	0.06	0.00	2000.00	0.80	0.91	1.04

Table 3.15: *Simulated posterior summary of reduced form of regression model by independent Metropolis algorithm using LaplacesDemon function.*

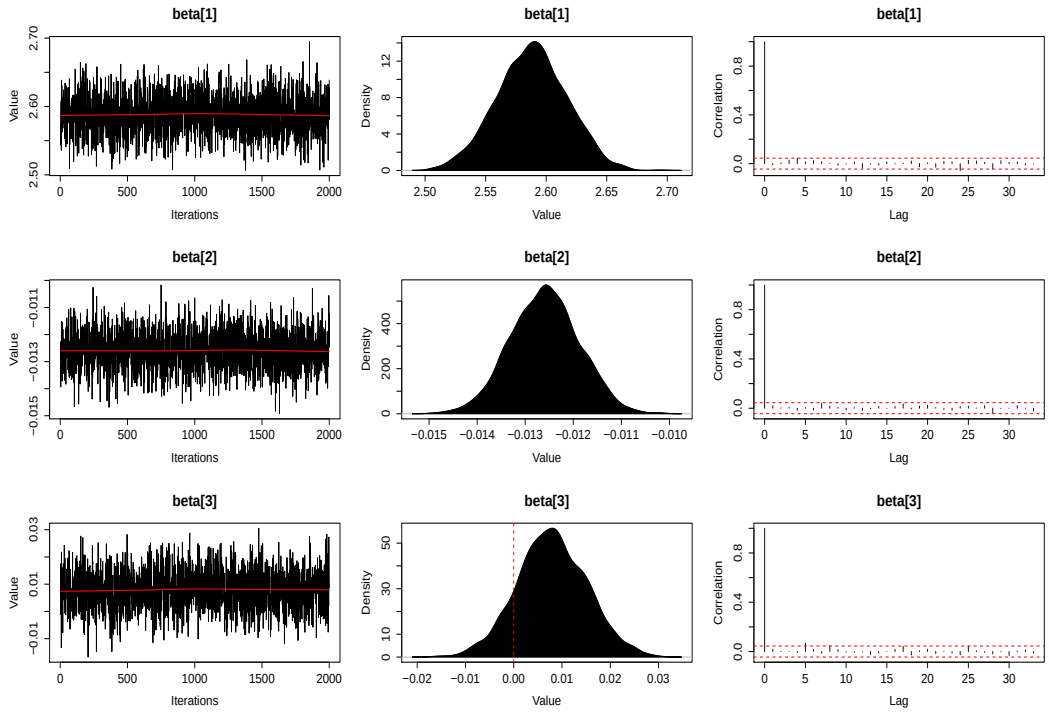

Figure 3.11: *Trace, posterior density and auto-correlation plots for variables of Burr type X distribution by independent Metropolis algorithm.*

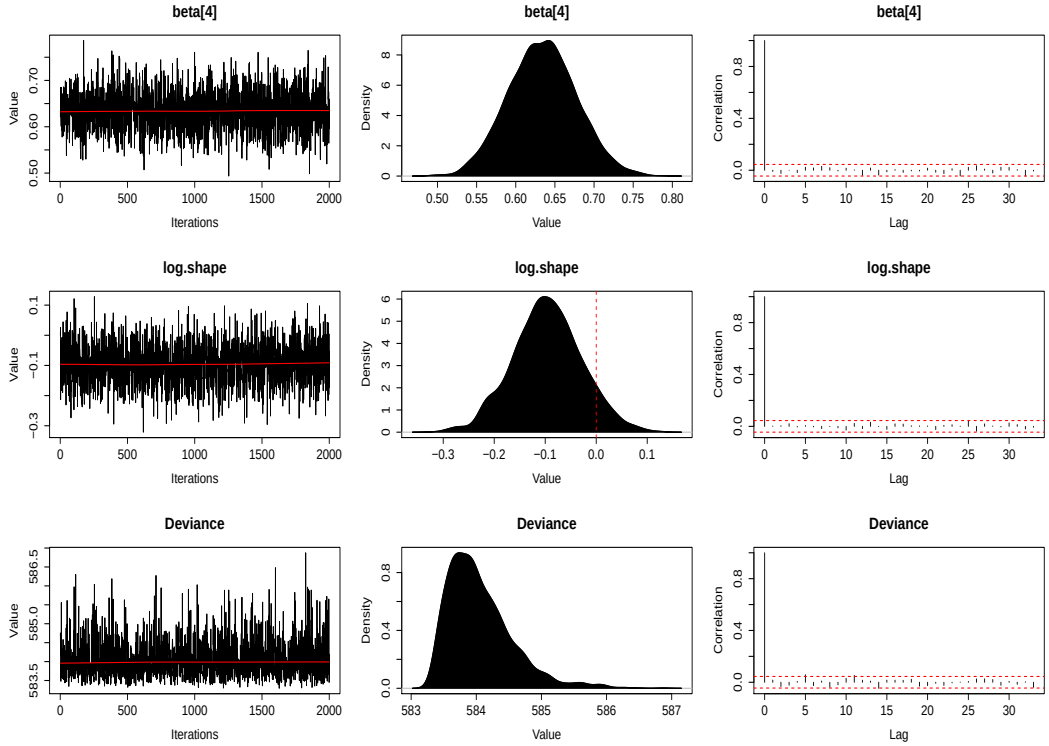

Figure 3.12: *Trace, posterior density and auto-correlation plots for variables of Burr type X distribution by independent Metropolis algorithm.*

3.13 Bayesian Modelling of reduced form of regression model for Rayleigh distribution

R code for the modeling of Rayleigh distribution in Bayesian framework are given below. Its approximated and simulated posterior summaries are listed in Table 3.16, 3.17 and 3.18, respectively. Also the graphical posterior summaries are reported in Figure 3.13.

```
y<-c(13,52,6,40,10,7,66,10,10,14,16,4,65,5,11,10,15,5,76,56,88,24,
   51,4,40,8,18,5,16,50,40,1,36,5,10,91,18,1,18,6,1,23,15,18,12,
   12,17,3)

censor<-c(1,0,1,1,1,0,1,0,1,1,1,1,1,1,0,1,0,1,0,0,1,1,1,1,0,1,1,1,
        1,1,1,1,1,1,1,1,0,1,0,1,1,1,1,1,0,1,1,0)

#####Blood urea nitrogen(Bun)#####
x1<-c(25,13,15,10,20,12,21,41,37,40,39,172,28,13,25,12,14,26,12,18,
    21,10,10,48,57,53,12,130,17,37,14,165,40,23,13,27,21,20,21,11,
    56,20,21,18,46,6,28,90)
x11<-x1-mean(x1)

##### Haemoglobin(Hb)####
x2<-c(14.6,12,11.4,10.2,13.2,9.9,12.8,14,7.5,10.6,11.2,10.1,6.6,
    9.7,8.8,9.6,13,10.4,14,12.5,14,12.4,10.1,6.5,12.8,8.2,14.4,
    10.2,10,7.7,5,9.4,11,9,14,11,10.8,5.1,13,5.1,11.3,14.6,8.8,
    7.5,4.9,5.5,7.5,10.2)
x21<-x2-mean(x2)

#### Bence-Jones protein####
x3<-c(1,0,1,1,0,0,1,1,0,0,0,1,0,0,0,0,0,0,0,0,1,0,1,0,1,0,0,0,0,
    1,0,0,1,0,0,1,0,1,0,0,0,0,0,1,0,0,0,1)

X<-cbind(1,x11,x21,x3)
J<-4
mon.names<-c("LP")
parm.names<-as.parm.names(list(beta=rep(0,J)))
MyData<-list(J=J,X=X,mon.names=mon.names,parm.names=parm.names,
        y=y,censor=censor)
```

```
Model<-function(parm,Data)
{
beta<-parm[1:Data$J]
beta.prior<-sum(dnorm(beta,0,1000,log=T))
mu<-tcrossprod(beta,Data$X)
scale<-exp(mu)
f1<-log(2)-log(scale)^2+log(y)-(y/scale)^2
s1<--(y/scale)^2
LL<-censor*f1+(1-censor)*s1
LL<-sum(LL)
LP<-LL+beta.prior
Modelout<-list(LP=LP,Dev=-2*LL,Monitor=c(LP),yhat=
          rray(length(y),scale),parm=parm)
return(Modelout)
}
Initial.Values<-c(coef(lm(log(y)~x11+x21+x3)))
M1<-LaplaceApproximation(Model,Initial.Values,Data=MyData,
    Sample=10000,Iterations=10000,Method="TR")
M1
Initial.Values<-as.initial.values(M2)
M2 <- LaplacesDemon(Model, Data=MyData,Initial.Values,
Covar=M2$Covar,Iterations=54000,Status=232,Thinning=27,
Algorithm="IM",Specs=list(mu=apply(M2$Posterior1,2,mean)))
```

	Mode	SD	LB	UB
Intercept	2.74	0.06	2.62	2.86
Bun	-0.01	0.00	-0.02	-0.01
Hb	0.05	0.02	0.02	0.08
Protein	0.46	0.10	0.26	0.66

Table 3.16: *Aprroximated posterior summary of reduced form of regression model of Rayleigh distribution.*

	Mean	SD	MCSE	ESS	LB	Median	UB
Intercept	2.75	0.06	0.00	10000.00	2.63	2.75	2.87
Bun	-0.01	0.00	0.00	10000.00	-0.02	-0.01	-0.01
Hb	0.05	0.02	0.00	10000.00	0.02	0.05	0.08
Protein	0.46	0.10	0.00	10000.00	0.26	0.46	0.66

Table 3.17: *Simulated posterior summary of reduced form of regression model by sampling importance resampling using the same function.*

	Mean	SD	MCSE	ESS	LB	Median	UB
Intercept	2.74	0.03	0.00	2000.00	2.69	2.74	2.79
Bun	-0.01	0.00	0.00	2000.00	-0.01	-0.01	-0.01
Hb	0.05	0.01	0.00	2000.00	0.04	0.05	0.06
Protein	0.47	0.04	0.00	2000.00	0.39	0.47	0.55

Table 3.18: *Simulated posterior summary of reduced form of regression model by independent Metropolis algorithm using LaplacesDemon function.*

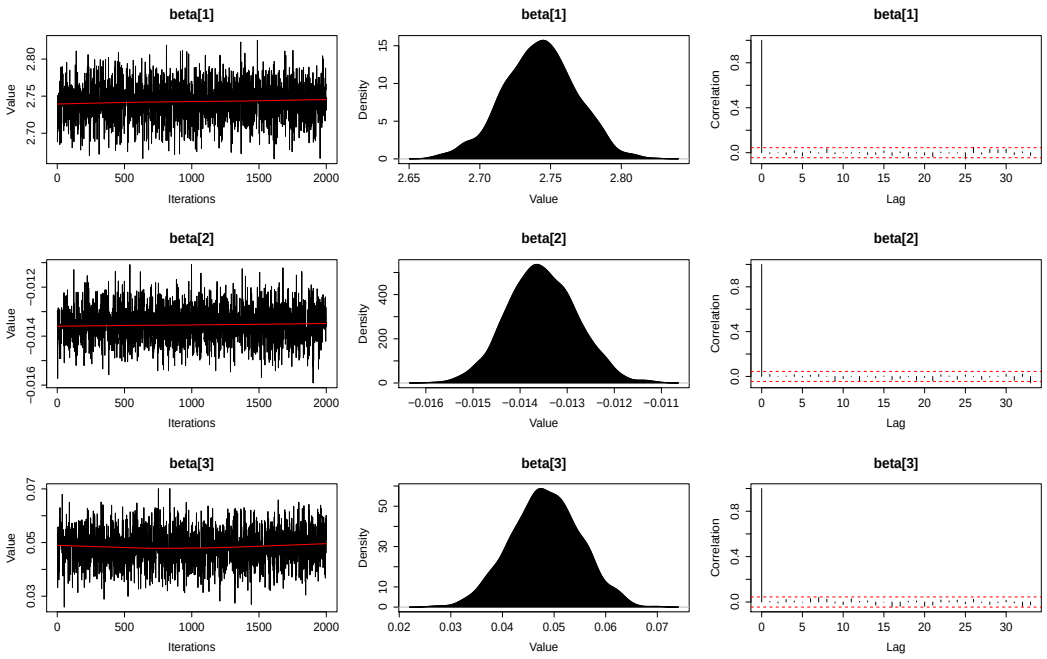

Figure 3.13: *Trace, posterior density and auto-correlation plots for variables of Rayleigh distribution by independent Metropolis algorithm.*

3.14 Bayesian Modelling of reduced form of regression model for Weibull distribution

```
library(LaplacesDemon)
y<-c(13,52,6,40,10,7,66,10,10,14,16,4,65,5,11,10,15,5,76,56,88,24,
   51,4,40,8,18,5,16,50,40,1,36,5,10,91,18,1,18,6,1,23,15,18,12,
   12,17,3)

censor<-c(1,0,1,1,1,0,1,0,1,1,1,1,1,1,0,1,0,1,0,0,1,1,1,1,0,1,1,1,
        1,1,1,1,1,1,1,1,0,1,0,1,1,1,1,1,0,1,1,0)

#####Blood urea nitrogen(Bun)#####
x1<-c(25,13,15,10,20,12,21,41,37,40,39,172,28,13,25,12,14,26,12,18,
    21,10,10,48,57,53,12,130,17,37,14,165,40,23,13,27,21,20,21,11,
    56,20,21,18,46,6,28,90)
x11<-x1-mean(x1)

##### Haemoglobin(Hb)####
x2<-c(14.6,12,11.4,10.2,13.2,9.9,12.8,14,7.5,10.6,11.2,10.1,6.6,
    9.7,8.8,9.6,13,10.4,14,12.5,14,12.4,10.1,6.5,12.8,8.2,14.4,
    10.2,10,7.7,5,9.4,11,9,14,11,10.8,5.1,13,5.1,11.3,14.6,8.8,
    7.5,4.9,5.5,7.5,10.2)
x21<-x2-mean(x2)

#### Bence-Jones protein####
x3<-c(1,0,1,1,0,0,1,1,0,0,0,1,0,0,0,0,0,0,0,0,1,0,1,0,1,0,0,0,0,
    1,0,0,1,0,0,1,0,1,0,0,0,0,0,1,0,0,0,1)

X<-cbind(1,x11,x21,x3)
J<-4
mon.names<-c("LP","shape")
parm.names<-as.parm.names(list(beta=rep(0,J),log.shape=0))
MyData<-list(J=J,X=X,mon.names=mon.names,parm.names=parm.names,
        y=y,censor=censor)
Initial.Values<-c(coef(lm(log(y)~x11+x21+x3)),log(1))
{
  #Parameters
  beta<-parm[1:Data$J]
```

```
  shape<-exp(parm[Data$J+1])
  # Log(Prior Densities)
  beta.prior<-sum(dnorm(beta,0,sqrt(1000),log=T))
  shape.prior<-dhalfcauchy(shape,20,log=T)
  # Loglikelihood
  mu<-tcrossprod(beta,Data$X)
  scale<-exp(mu)
  LL<-sum(censor*dweibull(Data$y,shape,scale,log=T)+
     (1-censor)*pweibull(Data$y,shape,scale,log.p=T,
     lower.tail=F))
  ## Log-posterior
  LP<-LL+beta.prior+shape.prior
  Modelout<-list(LP=LP,Dev=-2*LL,Monitor=c(LP,shape),yhat=
            rweibull(length(y),shape,scale),parm=parm)
  return(Modelout)
}
M1<-LaplaceApproximation(Model,Initial.Values,Data=MyData,
    Method="TR",Iterations=10000)
M1
Initial.Values <- as.initial.values(M2)
M2<-LaplacesDemon(Model,Data=MyData,Initial.Values,
Covar=M2$Covar,Iterations=24000,Status=154,Thinning=12,
Algorithm="IM",Specs=list(mu=apply(M2$Posterior1,2,mean)))
M2
```

	Mode	SD	LB	UB
Intercept	3.11	0.18	2.75	3.46
Bun	-0.02	0.00	-0.03	-0.01
Hb	0.08	0.05	-0.02	0.18
Protein	0.58	0.31	-0.04	1.21
log.shape	0.17	0.13	-0.08	0.43

Table 3.19: *Aprroximated posterior summary of reduced form of regression model of Weibull distribution.*

	Mean	SD	MCSE	ESS	LB	Median	UB
Intercept	3.15	0.20	0.01	1000.00	2.74	3.14	3.51
Bun	-0.02	0.01	0.00	1000.00	-0.02	-0.02	-0.00
Hb	0.09	0.05	0.00	1000.00	-0.01	0.09	0.22
Protein	0.61	0.36	0.01	1000.00	-0.07	0.62	1.35
shape	1.10	0.14	0.00	1000.00	0.86	1.09	1.40

Table 3.20: *Simulated posterior summary of reduced form of regression model by sampling importance resampling using the same function.*

	Mean	SD	MCSE	ESS	LB	Median	UB
Intercept	3.11	0.08	0.00	2000.00	2.96	3.11	3.26
Bun	-0.02	0.00	0.00	2000.00	-0.02	-0.02	-0.01
Hb	0.08	0.02	0.00	2000.00	0.04	0.08	0.12
Protein	0.59	0.13	0.00	1858.77	0.34	0.59	0.84
shape	1.17	0.06	0.00	2000.00	1.06	1.17	1.30

Table 3.21: *Simulated posterior summary of reduced form of regression model by independent Metropolis algorithm using LaplacesDemon function.*

The importance of MCMC tool can be realized in the analysis of reduced models. It is evident from these analysis that the summaries reported by SIR via `LaplaceApproximation` shows that some of the covariates are not significant whereas the summaries reported by `LaplacesDemon` is consistantly supporting the statistical significance of these covariates. We feel more comfortable with MCMC tools reported by `LaplacesDemon`.

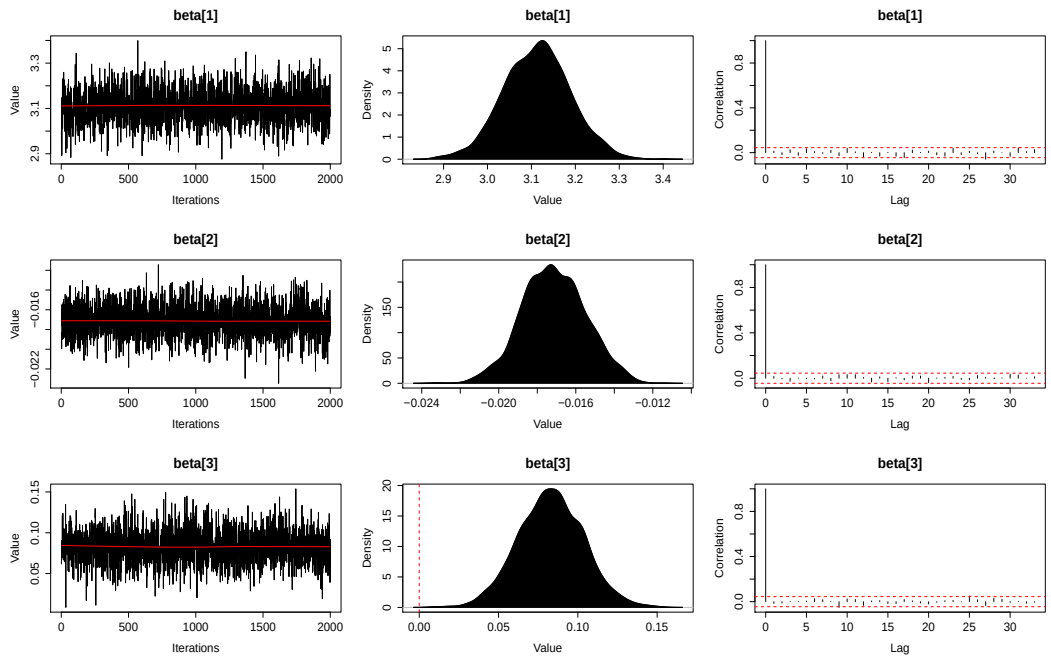

Figure 3.14: *Trace, posterior density and auto-correlation plots for variables of Weibull distribution by independent Metropolis algorithm.*

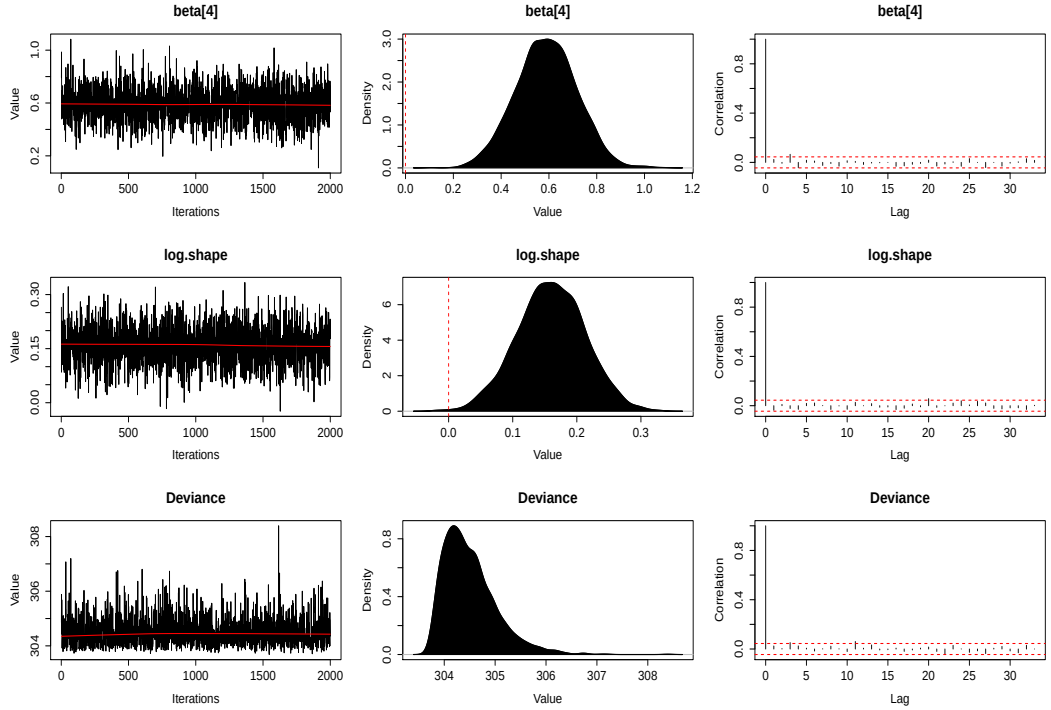

Figure 3.15: *Trace, posterior density and auto-correlation plots for variables of Weibull distribution by independent Metropolis algorithm.*

3.15 Model comparison

In the model selection problem, we must balance the complexity of a statistical model with its goodness of fit criterion. DIC and deviance appears to be a good choice if more than two models need to be compared and / or if it is important to reach a decision on which of the compared models to select. Table 3.22 shows the importance of exponentiated Weibull distribution in terms of fitting survival data. The difference in the values of DIC and deviance for EW and its sub-models EE is very small. Burr type X is also seems to be a good model after EW and EE. Overall, it could be seen that exponentiated Weibull model is the best choice for modelling such a survival data.

Models	Deviance	DIC
Exponentiated Weibull	256	258
Exponentiated exponential	257	258
Rayleigh	327	329
Weibull	306	307
Burr type X	259	260

Table 3.22: *Model comparison of exponentiated Weibull and its sub models for the given survival data. It is evident from this table that exponentieted Weibull fits much better than its sub models.*

CHAPTER 4

Bayesian Analysis of Lomax Family of Distributions

The Lomax distribution was first proposed by Lomax (1954). This distribution has been quite widely applied in a variety of fields such as actuarial science, medical and biological sciences, engineering, lifetime and reliability modeling (Hasan and Al-Ghamdi, 2009), although it was introduced originally for modeling business failure data. Furthermore, the data obtained from size distribution of computer files on servers (Holland et. al, 2006), income and wealth (Harris, 1968), receiver operating characteristic (ROC) curve analysis (Campbell, 1993), firm size (Corbellihi et. al, 2007) and Hirsch- related statistics (Glanzel, 2008) have been used for modeling using Lomax distribution. In the lifetime models, the Lomax model belongs to the family of decreasing failure rate by Chahkandi and Ganjali (2009). Many authors constructed generalizations of Lomax distribution. For example, Abdul-Moniem and Abdel-Hameed (2012) studied exponential Lomax (EL), Marshall-Olkin extended-Lomax (MOEL)by Ghitany et. al (2007) and Gupta et. al (2010), beta-Lomax (BL), Kumaraswamy-Lomax (KwL), McDonald-Lomax (McL) by Lemonte and Cordeiro (2013) and gamma-Lomax (GL) by Cordeiro et al. (2013). Recently, Tahir et. al (2015) introduced the Weibull Lomax (WL) distribution and studied its mathematical and statistical properties.

Many authors have worked on the estimation of Lomax distribution. The model exponential Lomax and Weibull Lomax are introduced recently in the literature in classical approach. This chapter includes the Bayesian modeling and illustration of these three distributions using real survival data.

In several pragmatic conditions, it could be noticed that the non-Bayesian analysis of such type of distributions is not an easy job, whereas it can be handled very effectively in a Bayesian scenario. Consequently, for the purpose of Bayesian analysis of Lomax, Weibull Lomax and exponential Lomax survival models, the two most far-reaching techniques, that is, optimization and simulation method are implemented using `LaplacesDemon` package of Statisticat LLC (2015). This package facilitates high-dimensional Bayesian inference, posing as its own intellect that have potential of impressive analysis, which is written entirely in R (R Core Team, 2015) aura and has an exceptional provision for user defined probability model.

4.1 The Lomax model

A random variable T has the Lomax distribution with two parameters α and λ, if it has probability density function (pdf) (for $T > 0$) given by

$$f(t;\alpha,\lambda) = \frac{\alpha}{\lambda}\left[1+\frac{t}{\lambda}\right]^{-(\alpha+1)} \qquad t>0, \qquad (\alpha,\lambda>0) \tag{4.1}$$

cumulative distribution function (cdf),

$$F(t;\alpha,\lambda) = 1-\left[1+\frac{t}{\lambda}\right]^{-\alpha} \qquad t>0, \qquad (\alpha,\lambda>0). \tag{4.2}$$

survival function,

$$S(t;\alpha,\lambda) = \left[1+\frac{t}{\lambda}\right]^{-\alpha} \qquad t>0, \qquad (\alpha,\lambda>0). \tag{4.3}$$

hazard function,

$$h(t;\alpha,\lambda) = \frac{\alpha}{\lambda}\left[1+\frac{t}{\lambda}\right]^{-1} \qquad t>0, \qquad (\alpha,\lambda>0). \tag{4.4}$$

4.1.1 Functions for Lomax distribution in R

1. R code for probability density function is

```
dlomax<-function(x,alpha,lambda){
alpha/lambda*(1+x/lambda)^(-(alpha+1))
}
```

2. R code for cumulative density function is

```
plomax<-function(x,alpha,lambda){
 1-(1+x/lambda)^(-alpha)
}
```

3. R code for random generation function is

```
rlomax<-function(n,alpha,lambda){
u<-runif(n)
x<-lambda*((1-u)^(-1/alpha)-1)
return(x)
}
```

4. R code for survival function is

```
survlomax<-function(x,alpha,lambda){
surv<-1-plomax(x,alpha,lambda)
return(surv)
}
```

5. R code for hazard function is

```
hlomax<-function(x,alpha,lambda){
haz<-dlomax(x,alpha,lambda)/slomax(x,alpha,lambda)
return(haz)
}
```

These functions `dlomax`, `plomax`, `survlomax` and `hlomax` have been used for the purpose of plotting. Figure 4.1 depicts the probability density, cumulative density, survival and hazard curves.

4.1.2 Construction of joint posterior distribution for Lomax model

Suppose that there are n subjects under study, and that associated with the ith individual is a survival time t_i and a censoring time t_{c_i} generated by Lomax distribution with parameters α and λ. The t's are assumed to be independent and identically distributed with density $f(t)$ and survival function $S(t)$. The exact survival time t_i of an individual will be observed only if $t_i \leq t_{c_i}$. The data in this framework can be represented by the n pairs of random variables (y_i, δ_i), where

$$y_i = min(t_i, t_{c_i})$$

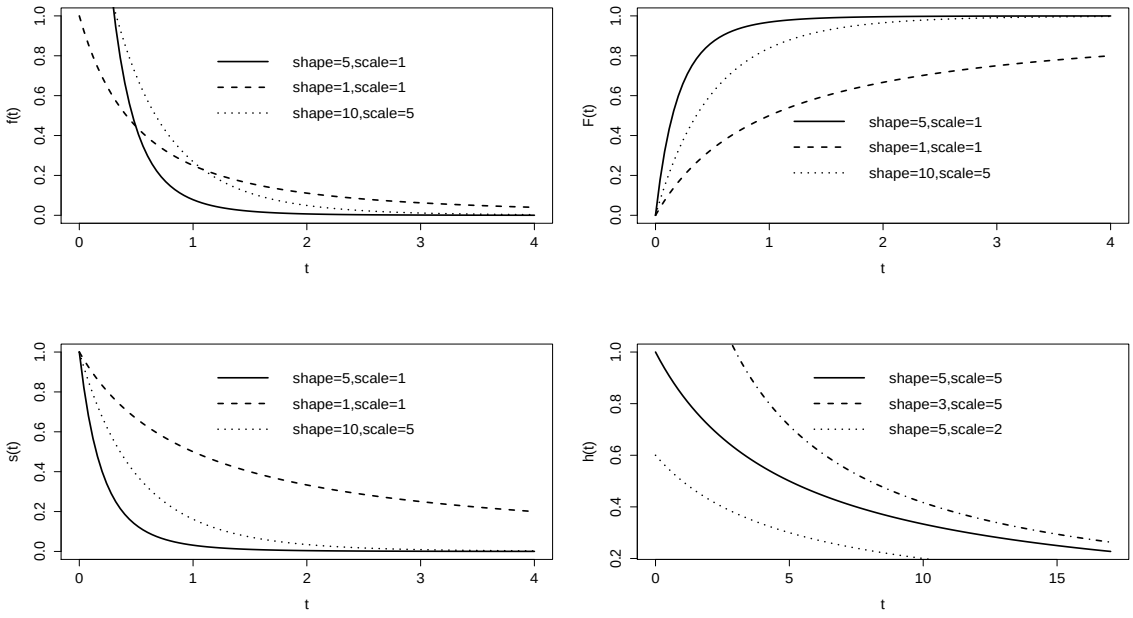

Figure 4.1: *Probability density plots, cdf, survival and hazard curves of Lomax distribution for different values of shape and scale.*

and

$$\delta_i = \begin{cases} 1 & \text{if } t_i \leq t_{c_i}, \\ 0 & \text{if } t_i > t_{c_i}. \end{cases} \tag{4.5}$$

Assuming a non-informative censoring mechanism, Lawless (2003), the likelihood and log-likelihood functions of Lomax distribution are given in Equation 4.6 and 4.7, respectively, by the following equation:

$$L \propto \prod_{i=1}^{n} [f(y_i)]^{\delta_i} [S(t_{c_i})]^{1-\delta_i}$$

Using Equation 4.1 and 4.3, the likelihood function is given by,

$$L(y;\alpha,\lambda) \propto \prod_{i=1}^{n} \left\{ \left[\frac{\alpha}{\lambda} \left(1 + \frac{y_i}{\lambda}\right)^{-(\alpha+1)} \right]^{\delta_i} \left[\left(1 + \frac{t_{c_i}}{\lambda}\right)^{-\alpha} \right]^{(1-\delta_i)} \right\} \tag{4.6}$$

After taking logarithm both side, we have

$$\log L(y;\alpha,\lambda) \propto \sum_{i=1}^{n} \left\{ \delta_i \log \left[\frac{\alpha}{\lambda} \left(1 + \frac{y_i}{\lambda}\right)^{-(\alpha+1)} \right] + (1-\delta_i) \log \left[\left(1 + \frac{t_{c_i}}{\lambda}\right)^{-\alpha} \right] \right\} \tag{4.7}$$

Considering weak-informative priors for the parameters of Lomax distribution. The prior assigned for parameter α is half-cauchy,

$$\alpha \sim \text{half-Cauchy}(\sigma)$$

which gives,

$$p(\alpha) = \frac{2\sigma}{\pi(\alpha^2 + \sigma^2)} \tag{4.8}$$

For the parameter $\lambda > 0$, a log-link function is used,

$$\log(\lambda) = X\beta$$

where, β is the vector of regression coefficient and can take any value on the real line, and X is the model matrix, or, equivalently,

$$\lambda = e^{X\beta}$$

For regression coefficient β, a normal prior distribution is assigned with parameter 0 and standard deviation 1000, that is,

$$\beta_j \sim N(0, 1000) \tag{4.9}$$

Following Equation 4.7, 4.8 and 4.9 and using Bayes theorem, the joint posterior distribution is given as,

$$p(\alpha, \beta | y, X) = \prod_{i=1}^{n} \left[\left\{ \frac{\alpha}{e^{x_i\beta}} \left(1 + \frac{y_i}{e^{x_i\beta}}\right)^{-(\alpha+1)} \right\}^{\delta_i} \left\{ \left(1 + \frac{t_{c_i}}{e^{x_i\beta}}\right)^{-\alpha} \right\}^{1-\delta_i} \right]$$
$$\left(\frac{2\sigma}{\pi(\alpha^2 + \sigma^2)}\right) \prod_{j=1}^{p} \left(\frac{1}{\sqrt{2\pi} * 1000} exp\left(\frac{-\beta_j^2}{2 * 1000^2}\right)\right) \tag{4.10}$$

Thus, marginal posterior of α

$$p(\alpha | y, X) = \int_{-\infty}^{\infty} p(\alpha, \beta | y, X) d\beta \tag{4.11}$$

and marginal posterior of β

$$p(\beta | y, X) = \int_{0}^{\infty} p(\alpha, \beta | y, X) d\alpha \tag{4.12}$$

4.2 The exponential Lomax model

A generalization to the Lomax distribution was suggested by (Abdul-Moniem, Abdel-Hameed, 2012) using Lehmann alternative type I proposed by Gupta et al. (1998). The three parameter EL pdf (for $t > 0$) is defined by

$$f(t; \nu, \alpha, \lambda) = \frac{\nu\alpha}{\lambda} \left(1 + \frac{t}{\lambda}\right)^{-(\alpha+1)} \left\{1 - \left(1 + \frac{t}{\lambda}\right)^{-\alpha}\right\}^{\nu-1} \tag{4.13}$$

cumulative distribution function (cdf),

$$F(t; \nu, \alpha, \lambda) = \left\{1 - \left(1 + \frac{t}{\lambda}\right)^{-\alpha}\right\}^{\nu}. \tag{4.14}$$

survival function,

$$s(t;\nu,\alpha,\lambda) = 1 - \left\{1 - \left(1 + \frac{t}{\lambda}\right)^{-\alpha}\right\}^{\nu}. \tag{4.15}$$

hazard function,

$$h(t;\nu,\alpha,\lambda) = \frac{\frac{\nu\alpha}{\lambda}\left(1+\frac{t}{\lambda}\right)^{-(\alpha+1)}\left\{1-\left(1+\frac{t}{\lambda}\right)^{-\alpha}\right\}^{\nu-1}}{1-\left\{1-\left(1+\frac{t}{\lambda}\right)^{-\alpha}\right\}^{\nu}}. \tag{4.16}$$

4.2.1 Functions for exponential Lomax in R

1. R code for probability density function is

```
dexplomax<-function(x,a,alpha,lambda){
a1<-(a*alpha)/lambda*(1+x/lambda)^(-(alpha+1))
a2<-(1-(1+x/lambda)^(-alpha))^(a-1)
return(a1*a2)
}
```

2. R code for cumulative density function is

```
pexplomax<-function(x,a,alpha,lambda){
a2<-(1-(1+x/lambda)^(-alpha))^a
return(a2)
}
```

3. R code for random generation function is

```
rexplomax<-function(n,alpha,a,lambda){
u<-runif(n)
x<-lambda*((1-u^(1/a))^(-1/alpha)-1)
return(x)
}
```

4. R code for survival function is

```
survexplomax<-function(x,a,alpha,lambda){
1-pexplomax(x,a,alpha,lambda)
}
```

5. R code for hazard function is

```
hexplomax<-function(x,a,alpha,lambda){
haz<-dexplomax(x,a,alpha,lambda)/survexplomax(x,a,alpha,lambda)
}
```

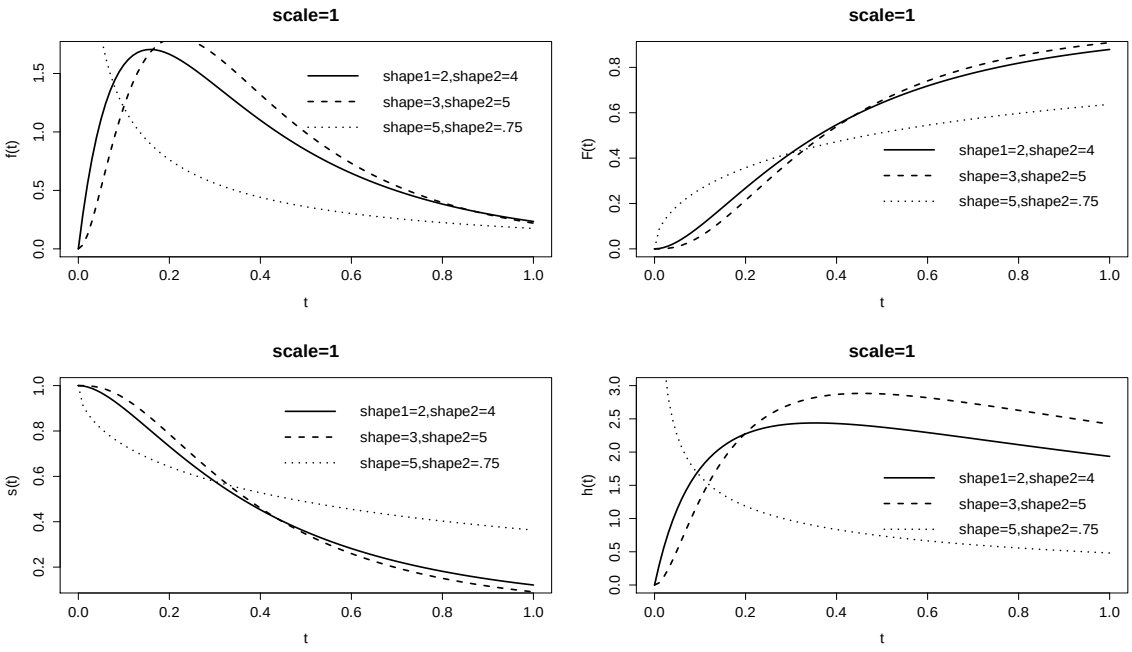

Figure 4.2: *Probability density plots, cdf, survival and hazard curves of exponential Lomax distribution for different values of shapes and at scale = 1.*

The joint posterior distribution of exponential Lomax distribution is,

$$p(\nu,\alpha,\beta|y,X)=\prod_{i=1}^{n}\left[\left\{\frac{\nu\alpha}{e^{x_i\beta}}\left(1+\frac{y_i}{e^{x_i\beta}}\right)^{-(\alpha+1)}\right\}^{\delta_i}\left\{\left(1-\left(1+\frac{t_{c_i}}{e^{x_i\beta}}\right)^{-\alpha}\right)^{\nu-1}\right\}^{1-\delta_i}\right]$$

$$\frac{2\gamma}{\pi(\alpha^2+\gamma^2)}\times\frac{2\gamma}{\pi(\nu^2+\gamma^2)}\times\prod_{j=1}^{p}\left\{\frac{1}{\sqrt{2\pi}*1000}\exp\left(\frac{-\beta_j^2}{2*1000^2}\right)\right\} \tag{4.17}$$

Thus, marginal posterior of α

$$p(\alpha|y,X)=\int_0^\infty\int_{-\infty}^\infty(\alpha,\nu,\beta|y,X)d\beta d\nu, \tag{4.18}$$

marginal posterior of β

$$p(\beta|y,X)=\int_0^\infty\int_0^\infty p(\alpha,\nu,\beta|y,X)d\alpha d\nu, \tag{4.19}$$

and marginal posterior of ν

$$p(\nu|y,X)=\int_{-\infty}^\infty\int_0^\infty p(\alpha,\nu,\beta|y,X)d\beta d\alpha \tag{4.20}$$

4.3 The Weibull Lomax model

A random variable T has the Weibull Lomax distribution with four parameters ν, η, α and λ, if it has probability density function (pdf) (for $t>0$) given by

$$f(t;\nu,\eta,\alpha,\lambda)=\frac{\nu\eta\alpha}{\lambda}\left(1+\frac{t}{\lambda}\right)^{\eta\alpha-1}\left[1-\left(1+\frac{t}{\lambda}\right)^{-\alpha}\right]^{\eta-1} exp\left\{-\nu\left\{\left[\left(1+\frac{t}{\lambda}\right)^{\alpha}-1\right]\right\}^{\eta}\right\} \tag{4.21}$$

cumulative distribution function (cdf),

$$F(t;\nu,\eta,\alpha,\lambda)=1-exp\left\{-\nu\left\{\left[\left(1+\frac{t}{\lambda}\right)^{\alpha}-1\right]\right\}^{\eta}\right\}. \tag{4.22}$$

survival function,

$$S(t;\nu,\eta,\alpha,\lambda)=exp\left\{-\nu\left\{\left[\left(1+\frac{t}{\lambda}\right)^{\alpha}-1\right]\right\}^{\eta}\right\}. \tag{4.23}$$

hazard function,

$$h(t)=\frac{\frac{\nu\eta\alpha}{\lambda}\left(1+\frac{t}{\lambda}\right)^{\eta\alpha-1}\left[1-\left(1+\frac{t}{\lambda}\right)^{-\alpha}\right]^{\eta-1}exp\left\{-\nu\left\{\left[\left(1+\frac{t}{\lambda}\right)^{\alpha}-1\right]\right\}^{\eta}\right\}}{exp\left\{-\nu\left\{\left[\left(1+\frac{t}{\lambda}\right)^{\alpha}-1\right]\right\}^{\eta}\right\}}. \tag{4.24}$$

4.3.1 Functions for Weibull Lomax in R

1. R code for probability density function is

```
dweiblomax<-function(x,a,b,alpha,lambda){
d1<-(a*b*alpha)/lambda*(1+x/lambda)^(b*alpha-1)
d2<-(1-(1+x/lambda)^(-alpha))^(b-1)
d3<-exp(-a*((1+x/lambda)^(alpha)-1)^b)
return(d1*d2*d3)
}
```

2. R code for cumulative density function is

```
pweiblomax<-function(x,a,b,alpha,lambda){
p<-1-exp(-a*((1+x/lambda)^(alpha)-1)^b)
return(p)
}
```

3. R code for random generation function is

```
rweiblomax<-function(n,a,b,alpha,lambda){
u<-runif(n)
x<-lambda*((-log(1-u)/a)^(1/b)+1)^(1/alpha)-1
return(x)
}
```

4. R code for survival function is

```
survweiblomax<-function(x,a,b,alpha,lambda){
surv<-1-pweiblomax(x,a,b,alpha,lambda)
return(surv)
}
```

5. R code for hazard function is

```
hweiblomax<-function(x,a,b,alpha,lambda){
haz<-dweiblomax(x,a,b,alpha,lambda)/survweiblomax(x,a,b,alpha,lambda)
return(haz)
}
```

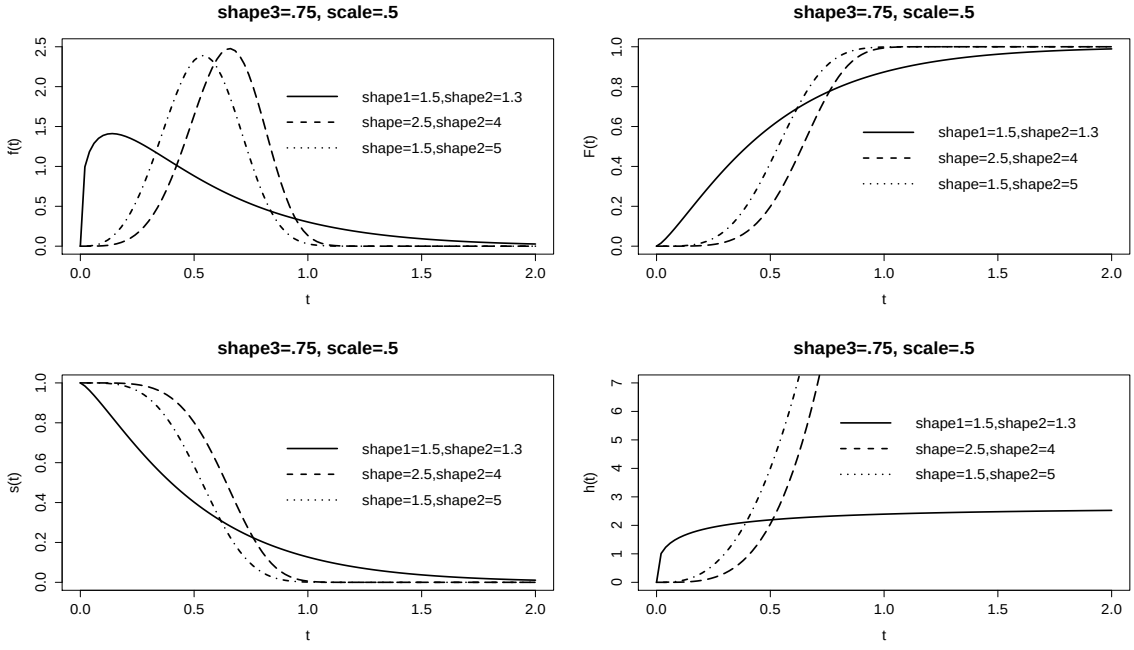

Figure 4.3: *Probability density plots, cdf, survival and hazard curves of Weibull Lomax distribution for different values of shapes and scale.*

The joint posterior distribution of Weibull Lomax is,

$$
\begin{aligned}
p(\nu,\eta,\alpha,\beta|y,X) = \prod_{i=1}^{n} \Bigg[& \left(\frac{\nu\eta\alpha}{e^{x_i\beta}}\left(1+\frac{y_i}{e^{x_i\beta}}\right)^{\eta\alpha-1}\left(1-\left(1+\frac{y_i}{e^{x_i\beta}}\right)^{-\alpha}\right)^{\eta-1}\right. \\
& exp\left(-\nu\left(\left(1+\frac{y_i}{e^{x_i\beta}}\right)^{\alpha}-1\right)^{\eta}\right)\Bigg)^{\delta_i} \\
& \left(exp\left(-\nu\left(\left(1+\frac{y_i}{e^{x_i\beta}}\right)^{\alpha}-1\right)^{\eta}\right)\right)^{1-\delta_i}\Bigg] \\
& \frac{2\gamma}{\pi(\alpha^2+\gamma^2)} \times \frac{2\gamma}{\pi(\nu^2+\gamma^2)} \times \frac{2\gamma}{\pi(\eta^2+\gamma^2)} \\
& \prod_{j=1}^{p}\left\{\frac{1}{\sqrt{2\pi}*1000}exp\left(\frac{-\beta_j^2}{2*1000^2}\right)\right\}
\end{aligned} \tag{4.25}
$$

Thus, marginal posterior of α

$$p(\alpha|y,X) = \int_0^\infty \int_0^\infty \int_{-\infty}^\infty p(\alpha,\nu,\beta|y,X)\,d\beta\,d\nu\,d\eta, \tag{4.26}$$

marginal posterior of β

$$p(\beta|y,X) = \int_0^\infty \int_0^\infty \int_0^\infty p(\alpha,\nu,\beta|y,X)\,d\alpha\,d\nu\,d\eta, \tag{4.27}$$

marginal posterior of ν

$$p(\nu|y,X) = \int_{-\infty}^\infty \int_0^\infty \int_0^\infty p(\alpha,\nu,\beta|y,X)\,d\beta\,d\alpha\,d\eta, \tag{4.28}$$

and marginal posterior of η

$$p(\eta|y,X) = \int_{-\infty}^\infty \int_0^\infty \int_0^\infty p(\alpha,\nu,\beta|y,X)\,d\beta\,d\alpha\,d\nu. \tag{4.29}$$

Since, Lomax, exponential Lomax and Weibull Lomax distributions contain two, three and four parameters respectively, hence, the evaluation of the joint posterior density which contains censoring mechanism also, will become a very difficult job. Consequently, some rigorous computational methods are required to solve the problem. To keep this in mind Tierney and Kadane (1989) suggested the use of Laplace approximation method. The Laplace approximation is a family of asymptotic techniques used to approximate integrals (Statisticat LLC 2015).

4.4 Survival data: veteran's administration lung cancer data

In this data, males with advanced inoperable lung cancer were randomized to either a standard or test chemotherapy. Only 9 of the 137 survival times were censored. The data is available in `survival` package and is presented in Kalbfleisch

and Prentice (1980, 2002). A portions of the data is analyzed by several other authors (Prentice,1973; Chen et al., 2002; Murphy et al., 1997; Bennett, 1983). In this analysis, the 137 subjects who completed the randomized portion of the trial and for whom complete covariate information was available are considered. Six covariates are available which include treatment, age, tumor cell type (adeno, small cell, squamous or large), time between initial diagnosis and enrollment in the trial, Karnofsky performance status, and prior therapy attempted (yes/no).

1. Treatment: **0** = standard, **1** = test.
2. Type of tumour: **1** = squamous, **2** = small cell, **3** = adeno, **4** = large cell.
3. Age in years.
4. Prior therapy: **0** = no, **1** = yes.
5. diagtime: Time in months from diagnosis to randomization.
6. Performance status: Karnofsky performance score (100=good).

4.5 Bayesian modeling of Lomax distribution

For Bayesian modeling of Lomax distribution on Veteran's adminitration lung cancer data involves the following steps:

1. Creation of lung cancer data.
2. Specification of model for Lomax distribution.
3. Generation of initial values
4. Fitting of Lomax distribution using `LaplaceApproximation` function for analytic approximation and then `LaplacesDemon` function for Markov chain Monte Carlo simulation.

By performing the above steps one by one, a complete posterior picture has been obtained by using two methods namely, Nelder-Mead optimization method for analytic approximation and independent Metropolis algorithm for simulation. Implementation has been made by using `LaplacesDemon` package. These steps would be discussed in the following sections.

4.5.1 Creation of lung cancer data

For illustrative purpose, a real survival data set called `veteran` that is provided with the `survival` package is used. The survival data, called, `veteran` contains six regressor variable i.e celltype, karno, diagtime, age, prior and trt, and its vector have been defined by objects names x_1, x_2, x_3, x_4, x_5 and x_6, respectively, using an extraction operator `$`.

```
library(LaplacesDemon)
library(survival)
data(veteran)
y<-veteran$time
x1<-veteran$karno
x2<-veteran$celltype
x3<-veteran$diagtime
x4<-veteran$age
x5<-veteran$prior
x6<-veteran$trt
censor<-veteran$status
N<-137
X<-cbind(1,x1,x2,x3,x4,x5,x6)
J<-7
```

X is called the model matrix which contains six columns of regressor variables and a column of $1's$ is also inserted into it as an intercept. Here $J = 7$, as X has seven columns.

```
mon.names<-c("LP","shape")
parm.names<-as.parm.names(list(beta=rep(0,J),log.shape=0))
MyData<-list(J=J,X=X,mon.names=mon.names,parm.names=parm.names,
        y=y,censor=censor)
```

Each parameter must have a name specified in the vector `parm.names`, and parameter names must be included with the data. The object is created by using the function `as.parm.names`. The object `mon.names` is meant for the variables to be monitored. Finally, lung cancer data in `LaplacesDemon` has been created with object name `MyData` which contains the list of `J`, `X`, `mon.names`, `parm.names`, `y`, and a vector of censored observation called `censor`.

4.5.2 Model specification for Lomax distribution

To use `LaplacesDemon` package, one must specify a model. Let's consider a regression model, which is often denoted as:

$$y \sim \text{Lomax}(\alpha, \lambda)$$

$$\log(\lambda) = X\beta$$

The response variable, y follow Lomax distribution with parameter shape and scale, the scale parameter λ is equal to the cross product of design matrix X and the parameter β.

`LaplacesApproximation` deterministically maximizes the logarithm of the unnormalized joint posterior density as specified in the `Model` function. In Bayesian inference, the logarithm of the unnormalized joint posterior density is proportional to the sum of the log-likelihood and logarithm of the prior densities:

$$\log[p(\theta|y)] \propto \log[p(y|\theta)] + \log[p(\theta)]$$

During each iteration `LaplacesApproximation` passes two arguments to `Model`: `parm` and `Data`. These arguments are specified in the beginning of the function (i.e first line of `Model`). After defining parameters of distribution the next step is the assigning of prior to them. To obtain log-likelihood, we need density function and survival function which are defined as f_1 and s_1, respectively. Then, the `Model` function is evaluated and the logarithm of the unnormalized joint posterior density is calculated as `LP`. This function returns an object called `Modelout`, which is a list of the objects, log-posterior `(LP)`, deviance = -2*log-likelihood `(Dev)`, monitored variable `(Monitor)`, `yhat` and `parm`.

```
Model<-function(parm,Data)
{
#Parameters
beta<-parm[1:Data$J]
shape<-exp(parm[Data$J+1])
# Log(Prior Densities)
beta.prior<-sum(dnorm(beta,0,sqrt(1000),log=T))
shape.prior<-dhalfcauchy(shape,20,log=T)
# Loglikelihood
mu<-tcrossprod(beta,Data$X)
scale<-exp(mu)
f1<-log(shape)-log(scale)-(shape+1)*log(1+y/scale)
```

```
s1<--shape*log(1+y/scale)
LL<-censor*f1+(1-censor)*s1
LL<-sum(LL)
## Log-posterior
LP<-LL+beta.prior+shape.prior
Modelout<-list(LP=LP,Dev=-2*LL,Monitor=c(LP,shape),yhat=
          rlomax(length(y),shape,scale),parm=parm)
return(Modelout)
}
```

4.5.3 Initial values

`LaplacesApproximation` requires a vector of initial values for the parameters. Each initial value is a starting point for the estimation of a parameter. Here, all initial values are set to zero and `LaplaceApproximation` function will optimize initial values using Nelder-Mead method.

```
Initial.Values<-c(rep(0,J),log(1))
```

However, we recommend better initial values obtain from fitting multiple regression model using logarithm of response variable.

```
Initial.Values<-c(coef(lm(log(y)~x1+as.numeric(x2)+x3+x4+
                as.numeric(x5)+x6)),log(1))
```

The performance of these initial values in terms of convergence could be seen in Section 4.5.5 through trace plots reported in Figure 4.5 and 4.6.

4.5.4 Fitting of data using LaplaceApproximation function

The `LaplaceApproximation` function deterministically maximizes the logarithm of the unnormalized joint posterior density with one of several optimization algorithms. The goal of Laplace approximation is to estimate the posterior mode and variance of each parameter. Here, an output object called `M1` will be created as a result of using the `LaplaceApproximation` function. The object `M1` gives two summaries, summary1 is obtained by Nelder-mead method and summary2 is obtained by sampling importance resampling method.

```
M1<-LaplaceApproximation(Model,Initial.Values,Data=MyData,
Method="NM",Iterations=100000)
```

4.5.5 Summarization of output

The first part of the Table 4.1 summarizes the point-estimated posterior modes. Uncertainty around the posterior mode is estimated from the asymptotic covariance matrix. Rows are parameters i.e celltype, karno, diagtime, age, prior and treatment. The following columns are included: Mode, SD (Standard Deviation), LB (Lower Bound), and UB (Upper Bound). The bounds constitute a 95% credible interval.

The second part of the Table 4.1 summarizes the posterior samples drawn with sampling importance resampling (SIR) when sir=TRUE, given the point-estimated posterior mode and the covariance matrix obtain from `LaplaceApproximation`. Again rows are parameters. The following columns are included: Mean, SD (Standard Deviation), LB (Lower Bound), and UB (Upper Bound). The bounds constitute a 95% credible interval.

Optimization- Nelder and Mead				
	Mode	SD	LB	UB
Intercept	4.659	1.231	2.198	7.121
Karno	0.037	0.005	0.026	0.048
Celltype	-0.103	0.088	-0.279	0.072
Diagtime	-0.001	0.009	-0.018	0.017
Age	0.003	0.010	-0.017	0.023
Prior therapy	0.009	0.023	-0.038	0.056
Treatment	-0.153	0.202	-0.557	0.251
log.shape	1.967	0.682	0.602	3.332
Simulation- Sampling Importance Resampling				
	Mean	SD	LB	UB
Intercept	4.601	1.221	2.305	7.127
Karno	0.042	0.011	0.037	0.058
Cell type	-0.099	0.095	-0.281	0.089
Diagnosis time	0.002	0.013	-0.022	0.021
Age	0.004	0.011	-0.025	0.028
Prior therapy	0.012	0.024	-0.048	0.061
Treatment	-0.161	0.213	-0.557	0.264
shape	8.651	6.755	2.443	29.858

Table 4.1: *Posterior summaries of the lung cancer data using the function* ***`LaplaceApproximation`****, which is based on asymptotic approximation theory, and posterior summary due to sampling importance resampling method respectively, using the same function*

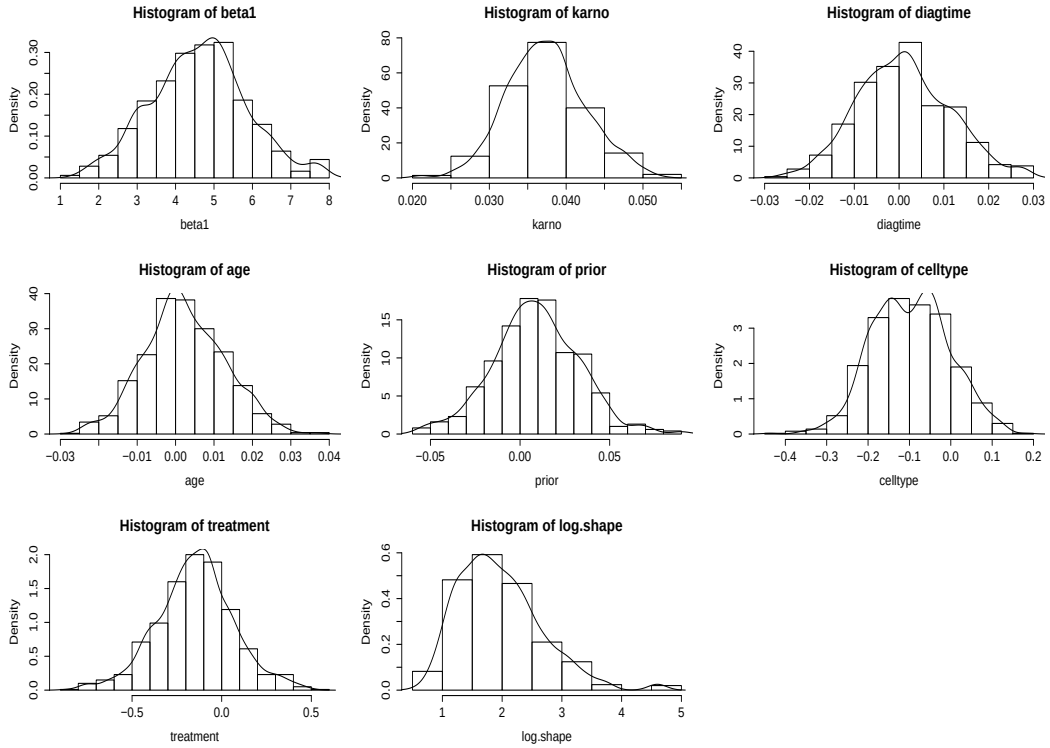

Figure 4.4: *Histograms and posterior densities of all the parameters and regression coefficients $\boldsymbol{\beta's}$ for Lomax distribution.*

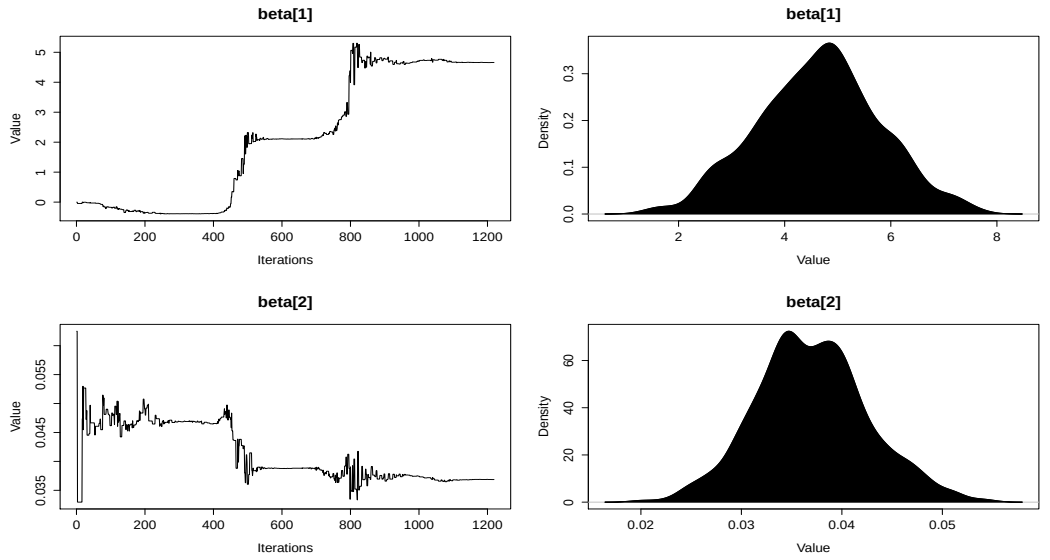

Figure 4.5: *Trace and posterior density plots of Lomax distribution at inital values zero for all the parameters.*

In Figure 4.5 convergence starts from 1000th iteration whereas in Figure 4.6 algorithm converges at 600th iteration. Hence, it shows 40% increase in the efficiency when initial values are obtained by using function `lm` as compared to the initial values taken as zero.

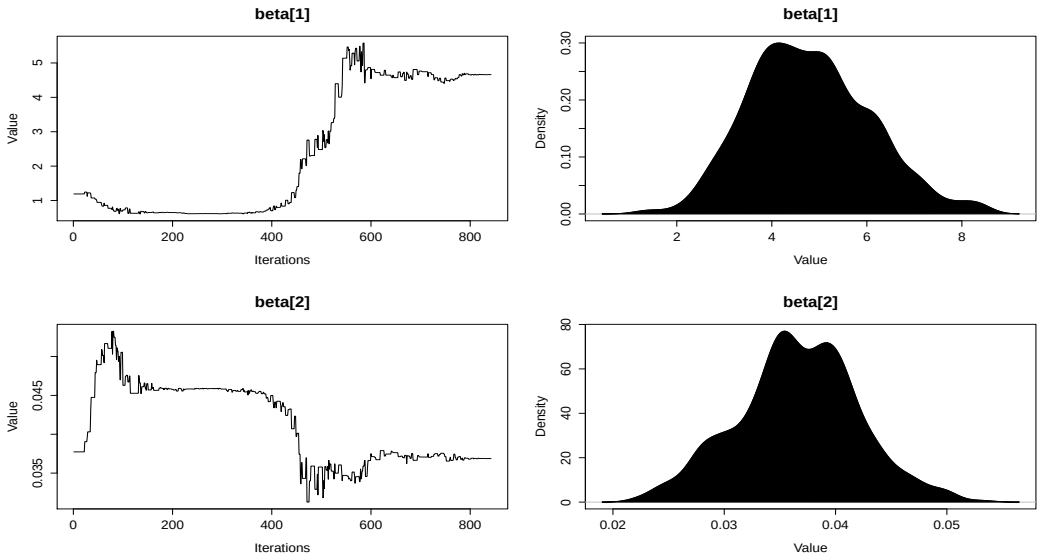

Figure 4.6: *Trace and posterior density plots of Lomax distribution at inital values obtain from fitting multiple regression model using logarithm of response variable.*

4.5.6 Fitting of data using LaplacesDemon function

Now, we have to explore the same veteran data using function `LaplacesDemon`. The `LaplacesDemon` function is the main function of `LaplacesDemon` package. It maximizes the logarithm of the unnormalized joint posterior density with MCMC and provides samples of the marginal posterior distributions, deviance, and other monitored variables. In `LaplacesDemon` function there is an argument called `Algorithm`, here the algorithm used for simulation from joint posterior distribution is independent-Metropolis algorithm. Multivariate normal has been treated as a proposal distribution $q(\theta)$. Here, the proposal distribution does not depend on the previous state of the chain. The IM algorithm is efficient when the proposal is a good approximation of the target posterior distribution. Good independent proposal densities can be based on `LaplaceApproximation` (Tierney and Kadane, 1986; Tierney et al., 1989; Erkanli, 1994). Thus, a generally successful proposal can be obtained by a multivariate normal distribution with mean equal to the posterior mode and precision matrix

$$H(\hat{\theta}) = \left(-\frac{\partial^2 log p(\theta|y)}{\partial\theta_i \partial\theta_j} |_{\theta=\hat{\theta}} \right)$$

that is, minus the second derivative matrix of the log-posterior density

$$\log p(\theta|y) = \log p(y|\theta) + \log p(\theta)$$

evaluated at the posterior mode $\hat{\theta}$. Consequently, an efficient proposal is given by,

$$q(\theta) = N\left(\hat{\theta}, [H(\hat{\theta})]^{-1}\right)$$

The posterior mode can be evaluated by some of the efficient methods provided in `LaplaceApproximation` with object `M1`. Among the optimization methods the performance of Nelder-Mead (1965) seems to be the best. Thus object `M1` is created by making use of `LaplaceApproximation` with the choice of optimization method of Nelder-Mead "N-M". When low information prior is used, then an adequate proposal can be obtained by setting the mean equal to the corresponding maximum-likelihood estimator (MLE) and the precision equal to its observed Fisher information matrix.
The acceptance probability, when proposing a transition from θ to θ' , is given by

$$\alpha = \min\left(1, \frac{p(\theta'|y)q(\theta)}{p(\theta|y)q(\theta')}\right)$$

which can be reexpressed as,

$$\alpha = \min\left(1, \frac{w(\theta')}{w(\theta)}\right),$$

where $w(\theta) = p(\theta|y)/q(\theta)$ is the ratio between the target and the proposal distribution and is equivalent to the importance weight used in importance sampling (Ntzroufras, 2009). This theory is implement in `LaplacesDemon` with object name `M2`. `M2` is an object of class demonoid, which means that since it has been assigned a customized class, other functions have been custom-designed to work with it. This `M2` object contains an argument, called `Covar`, the covariance matrix may be input from the `LaplaceApproximation` function `M1`. Another argument is `Specs`, which accepts the list of specifications for the MCMC algorithm.

```
Initial.Values<-as.initial.values(M1)
M2<-LaplacesDemon(Model, Data=MyData, Initial.Values,
    Covar=M1$Covar, Iterations=2000, Status=F, Thinning=1,
    Algorithm="IM",Specs=list(mu=M1$Summary1[1:length(Initial.Values),1]))
M2
```

4.5.7 Output by simulations

The `LaplacesDemon` function also generates two posterior summaries. `Summary1` gives the the marginal posterior distributions of the parameters, deviance, and monitored variables. The following summary statistics are included: mean, standard deviation, MCSE (Monte Carlo Standard Error), ESS which is the effective sample size due to autocorrelation, and finally the 2.5%, 50%, and 97.5% quantiles are reported in Table 4.2. MCSE is essentially a standard deviation around the marginal posterior mean that is due to uncertainty associated with using MCMC. `Summary2` is identical to `Summary1`, except that it is calculated only on the stationary samples and it ensures that convergence has been reached its equilibrium distribution. The

	Mean	SD	MCSE	ESS	LB	Median	UB
Intercept	4.529	0.475	0.012	2000.00	3.655	4.576	5.576
Karno	0.042	0.001	0.0023	2000.00	0.034	0.042	0.044
Cell type	-0.101	0.043	0.001	2000.00	-0.175	-0.101	-0.034
Diagnosis time	-0.002	0.003	0.002	2000.00	-0.013	-0.001	0.014
Age	0.003	0.001	0.002	2000.00	-0.001	0.001	0.034
Prior therapy	0.021	0.011	0.001	2000.00	-0.013	0.011	0.034
Treatment	-0.167	0.083	0.003	2000.00	-0.321	-0.174	-0.023
shape	7.585	2.086	0.051	2000.00	4.373	7.284	12.285

Table 4.2: *Simulated posterior summary of lung cancer data by independent Metropolis algorithm under the assumption of Lomax model.*

convergence of the algorithm can also be determined by MCSE, Table 4.2 shows very small values of this error which indicates that we have calculated the quantity of interest with high precision. From Figure 4.7, convergence can also be monitored through the trace and autocorrelation plots. Trace plots in the leftmost panel are very much convincing in terms of convergence as all generated values within a parallel zone. Monitoring autocorrelation is also very useful as it is evident from rightmost panel of the Figure 4.7 that the low values indicate fast convergence.

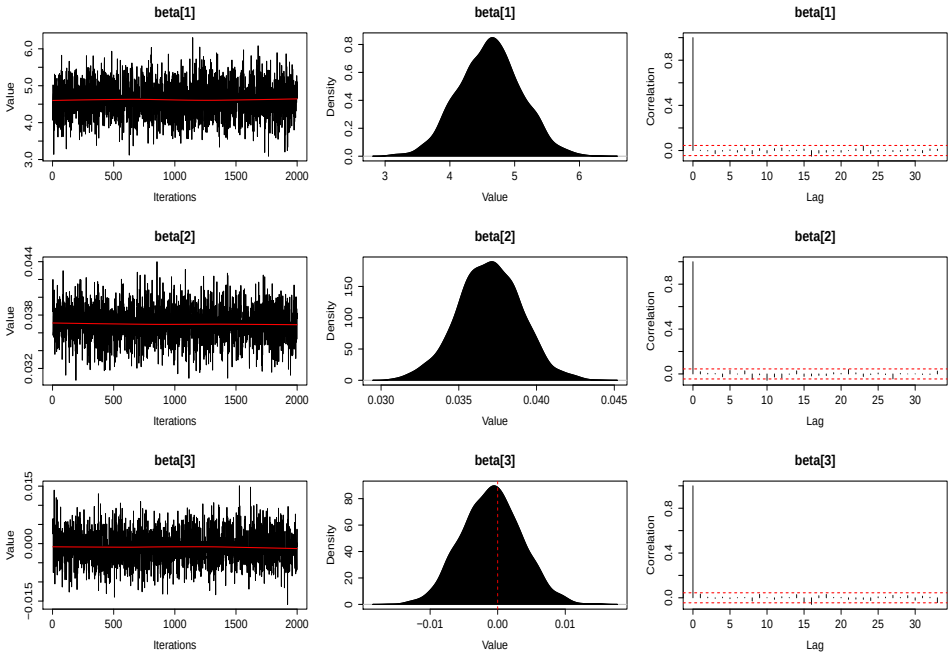

Figure 4.7: *Simulated posterior density plots of the parameter of Lomax distribution. The leftmost is the trace plot, the middlemost is the density plot and the rightmost is the auto correlation plot, showing low autocorrelation at different lags.*

4.6 Bayesian Modeling of exponential Lomax distribution

4.6.1 Creation of lung cancer data for exponential Lomax distribution

In this section data for exponential Lomax has been created with object name `MyData` which contains the list of vectors, that are, model matrix `X`, survival time vector `y`, monitoring variables `mon.names`, list of parameters of the model `parm.names`, vector of censored observations `censor`. `R` commands for the creation of veteran's lung cancer data for exponential Lomax distribution are described below:

```
y<-veteran$time
x1<-veteran$karno
x2<-veteran$celltype
x3<-veteran$diagtime
x4<-veteran$age
x5<-veteran$prior
x6<-veteran$trt
```

```
censor<-veteran$status
N<-137
X<-cbind(1,x1,x2,x3,x4,x5,x6)
J<-7
mon.names<-c("LP","shape1","shape2")
parm.names<-as.parm.names(list(beta=rep(0,J),log.shape1=0,
            log.shape2=0))
MyData<-list(J=J,X=X,mon.names=mon.names,parm.names=
        parm.names,y=y,censor=censor)
```

4.6.2 Specification of model for exponential Lomax distribution

Let's consider a regression model, which can be written as:

$$y \sim \boldsymbol{EL}(\nu, \alpha, \lambda)$$

where, ν and α are the two shape parameters and λ is the scale parameter. Also,

$$\log(\lambda) = X\beta$$

prior for ν,

$$\nu \sim \text{half-Cauchy}(\mathbf{25})$$

prior for α,

$$\alpha \sim \text{half-Cauchy}(\mathbf{25})$$

prior for β,

$$\beta_j \sim \boldsymbol{N}(0, 1000)$$

All these parameters and their priors have been defined in the function called `Model`. For exponential Lomax distribution there is no distribution function available in this package, so we will define density and survival function of EL distribution with object name `f1` and `s1`, respectively. The `R` command for the model specification of exponential Lomax distribution is given below:

```
Model<-function(parm,Data)
{
#Parameters
beta<-parm[1:Data$J]
shape1<-exp(parm[Data$J+1])
```

```
shape2<-exp(parm[Data$J+2])
# Log(Prior Densities)
beta.prior<-sum(dnorm(beta,0,sqrt(1000),log=T))
shape1.prior<-dhalfcauchy(shape1,20,log=T)
shape2.prior<-dhalfcauchy(shape2,20,log=T)
# Loglikelihood
mu<-tcrossprod(beta,Data$X)
scale<-exp(mu)
f1<-log(shape1)+log(shape2)-log(scale)-
    (shape2+1)*log(1+y/scale)+(shape1-1)*
    log(1-(1+y/scale)^(-shape2))
   s1<-log(1-(1-(1+y/scale)^(-shape2))^shape1)
  LL<-censor*f1+(1-censor)*s1
  LL<-sum(LL)
  ## Log-posterior
  LP<-LL+beta.prior+shape1.prior+shape2.prior
  Modelout<-list(LP=LP,Dev=-2*LL,Monitor=c(LP,shape1,shape2),
            yhat=rexplomax(length(y),shape1,shape2,scale),
            parm=parm)
  return(Modelout)
}
```

4.6.3 Fitting of the data using function LaplaceApproximation

Let us fit the model using `LaplaceApproximation` with the option of Nelder-Mead (1965) method of optimization as,

```
Initial.Values<-c(coef(lm(log(y)~x1+as.numeric(x2)+x3+x4+
                as.numeric(x5)+x6)),log(1),log(1))
M1<-LaplaceApproximation(Model,Initial.Values,Data=MyData,
    Samples=5000,Method="NM",Iterations=10000)
```

4.6.4 Output summary

The output obtained by `LaplaceApproximation` is being reported in Table 4.3. This contains the the posterior mode, posterior mean, posterior sd, **2.5%** and **97.5%** quantiles.

Optimization- Nelder and Mead Method				
	Mode	SD	LB	UB
Intercept	2.077	1.219	-0.362	4.516
Karno	0.040	0.005	0.030	0.049
Cell type	-0.010	0.091	-0.191	0.172
Diagnosis time	0.002	0.009	-0.017	0.021
Age	0.008	0.010	-0.011	0.028
Prior therapy	0.004	0.022	-0.040	0.047
Treatment	-0.095	0.190	-0.476	0.286
log.shape1	0.576	0.230	0.117	1.036
log.shape2	0.902	0.356	0.190	1.614
Simulation-Sampling Importance Resampling				
	Mean	SD	LB	UB
Intercept	2.131	1.302	-0.420	4.501
Karno	0.040	0.005	0.033	0.046
Cell type	-0.011	0.100	-0.194	0.196
Diagnosis time	0.001	0.014	-0.023	0.025
Age	0.011	0.010	-0.014	0.035
Prior	0.001	0.023	-0.056	0.055
Treatment	-0.115	0.206	-0.509	0.268
shape1	2.026	0.537	1.285	3.488
shape2	2.615	1.042	1.371	5.565

Table 4.3: *Posterior summaries of lung cancer data by LaplaceApproximation function, giving two summaries first from Nelder-mead method and second is from sampling importance resampling under the assumption of exponential Lomax model.*

4.6.5 Fitting of the data using LaplacesDemon function

Let us fit the model for the same data using `LaplacesDemon` with the option of independent Metropolis algorithm of simulation. The output obtained by simulation is reported in Table 4.4 and the posterior density plots are reported in Figure 4.9. There are six regressors but only plots of two regressors are reported (Karno=beta[2], Cell type=beta[3] and beta[1] is the intercept). In Figure 4.9 all the three rows are considered as beta[1], beta[2] and beta[3], respectively. These plots shows the well mixing of the chain and low auto-correlation shows fast convergence of algorithm for all the variables.

```
Initial.Values<-as.initial.values(M1)
```

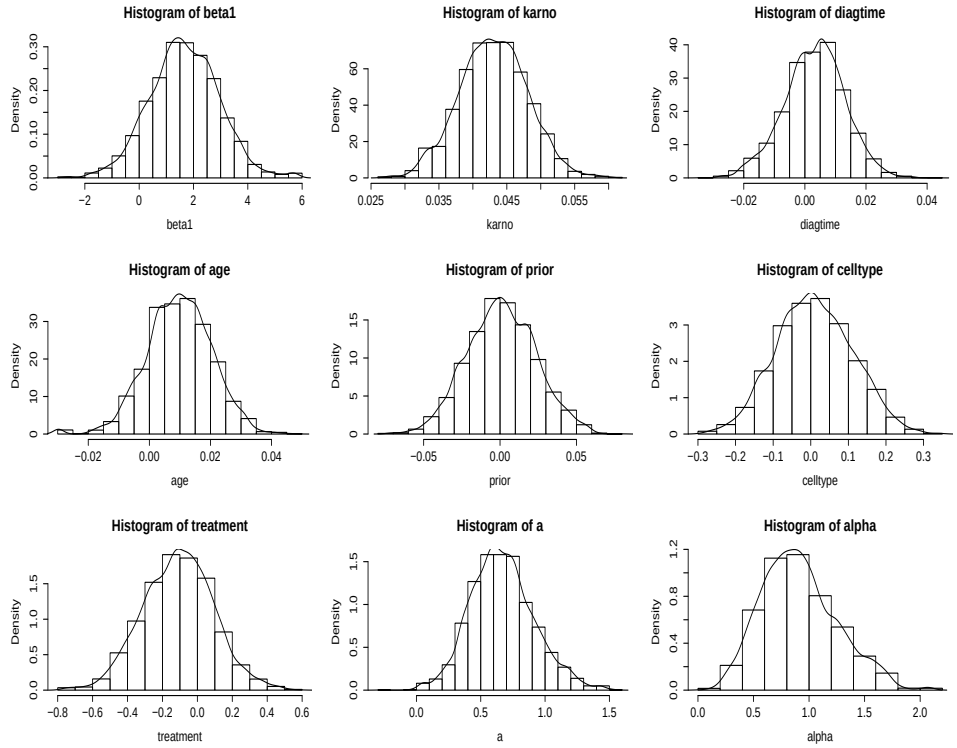

Figure 4.8: *Histogram and posterior densities of all the parameters and regression coefficients* $\boldsymbol{\beta's}$ *under the assumption of exponential Lomax distribution.*

```
M2<-LaplacesDemon(Model, Data=MyData, Initial.Values,
    Covar=M1$Covar, Iterations=2000, Status=F, Thinning=1,
    Algorithm="IM",Specs=list(mu=M1$Summary1[1:length(Initial.Values),1]))
```

	Mean	SD	MCSE	ESS	LB	Median	UB
Intercept	1.285	0.462	0.012	2000.00	0.363	1.309	2.176
Karno	0.039	0.001	0.002	1641.23	0.041	0.044	0.053
Cell type	0.034	0.041	0.003	2000.00	-0.052	0.031	0.109
Diagnosis time	0.002	0.001	0.003	2000.00	-0.005	0.008	0.016
Age	0.01	0.001	0.002	2000.00	0.012	0.016	0.024
Prior therapy	0.004	0.010	0.004	2000.00	-0.018	0.004	0.028
Treatment	-0.092	0.080	0.001	1543.30	-0.241	-0.095	0.078
shape1	2.071	0.203	0.001	1788.54	1.714	2.055	2.492
shape2	2.383	0.300	0.011	2000.00	1.841	2.364	3.028

Table 4.4: *Simulated posterior summaries obtained by independent Metropolis algorithm.*

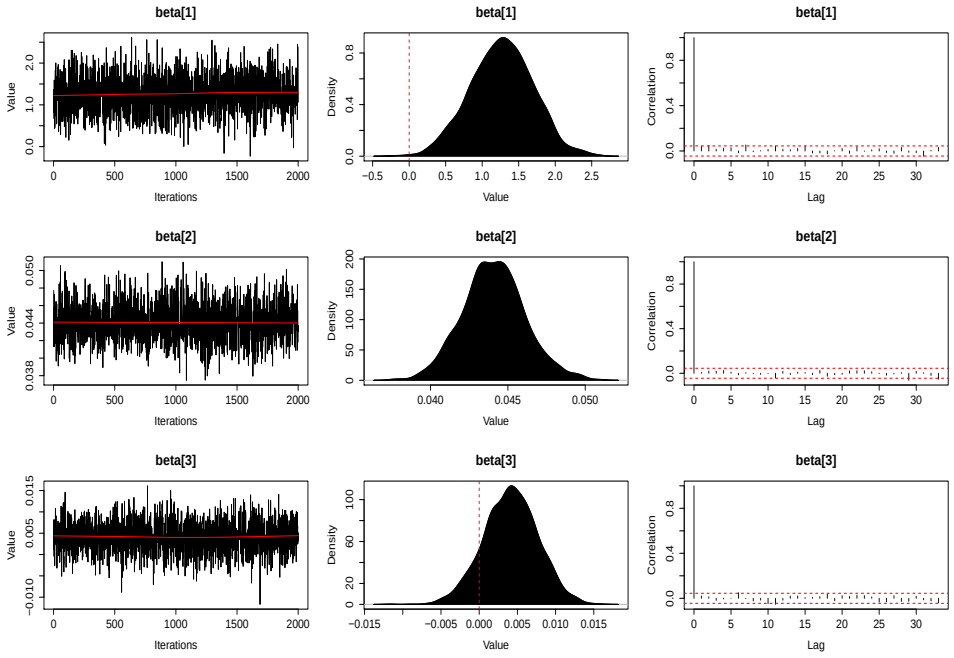

Figure 4.9: *Simulated posterior density plots of the parameters of the distribution. The leftmost is the trace plot, the middlemost is the density plots and the rightmost is the auto correlation plots, showing low autocorrelation at different lags.*

4.7 Bayesian Modeling of Weibull Lomax distribution

Weibull Lomax distribution has four parameters in which three are shape parameters and one is scale. In this section, veteran's lung cancer data has been created with object name `MyData`. Since this distribution has three shape parameters, so the function `Model` contains the definition of parameters as `shape1`, `shape2` and `shape3` and for regression coefficient it is defined as `beta`. Then, half-Cauchy prior distribution has been assigned for all the three shape parameters and normal prior is for `beta`. After that the density and survival function of Weibull Lomax distribution has been defined as f1 and s1, respectively. All these informations have been gathered together to calculate `LP`. Object `M1` has been assigned for the implementation of `LaplaceApproximation`. `R` code for Bayesian modelling of Weibull Lomax distribution in `LaplacesDemon` package have been describe below:

```
y<-veteran$time
x1<-veteran$karno
x2<-veteran$celltype
x3<-veteran$diagtime
x4<-veteran$age
```

```
x5<-veteran$prior
x6<-veteran$trt
censor<-veteran$status
N<-137
X<-cbind(1,x1,x2,x3,x4,x5,x6)
J<-7
mon.names<-c("LP","shape1","shape2","shape3")
parm.names<-as.parm.names(list(beta=rep(0,J),log.shape1=0,
            log.shape2=0,log.shape3=0))
MyData<-list(J=J,X=X,mon.names=mon.names,parm.names=
        parm.names,y=y,censor=censor)
Initial.Values<-c(coef(lm(log(y)~x1+as.numeric(x2)+x3+x4+
                as.numeric(x5)+x6)),log(1),log(1),log(1))

Model<-function(parm,Data)
{
#Parameters
beta<-parm[1:Data$J]
shape1<-exp(parm[Data$J+1])
shape2<-exp(parm[Data$J+2])
shape3<-exp(parm[Data$J+3])
# Log(Prior Densities)
beta.prior<-sum(dnorm(beta,0,sqrt(1000),log=T))
shape1.prior<-dhalfcauchy(shape1,20,log=T)
shape2.prior<-dhalfcauchy(shape2,20,log=T)
shape3.prior<-dhalfcauchy(shape3,20,log=T)
# Loglikelihood
mu<-tcrossprod(beta,Data$X)
scale<-exp(mu)
f1<-log(shape1)+log(shape2)+log(shape3)-log(scale)+
    (shape2*shape3-1)*log(1+y/scale)+(shape2-1)*
    log(1-(1+y/scale)^(-shape3))-shape1*
    (((1+y/scale)^shape3-1))^shape2
s1<--shape1*(((1+y/scale)^shape3-1))^shape2
LL<-censor*f1+(1-censor)*s1
LL<-sum(LL)
## Log-posterior
LP<-LL+beta.prior+shape1.prior+shape2.prior+shape3.prior
Modelout<-list(LP=LP,Dev=-2*LL,Monitor=c(LP,shape1,shape2,shape3),
```

```
        yhat=rweiblomax(length(y),shape1,shape2,shape3,scale),
        parm=parm)
return(Modelout)
}
M1<-LaplaceApproximation(Model,Initial.Values,Data=MyData,
    Method="NM",Iterations=100000)
```

The output obtained by object `M1` is listed in Table 4.5 and its graphical summaries in terms of histograms are reported in Figure 4.10.

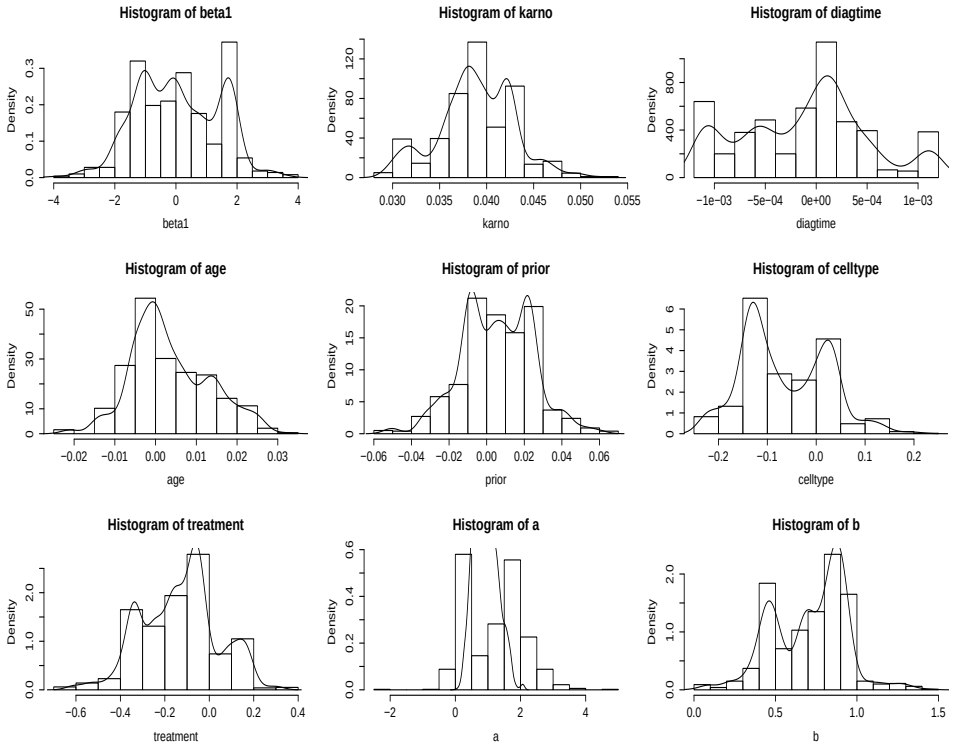

Figure 4.10: *Histograms and posterior densities of all the parameters and regression coefficients* $\boldsymbol{\beta's}$ *for Weibull Lomax distribution.*

Optimization- Nelder and Mead Method				
	Mean	SD	LB	UB
Intercept	1.239	1.707	-2.174	4.653
Karno	0.037	0.005	0.027	0.047
Cell type	-0.082	0.089	-0.260	0.095
Diagnosis time	-0.001	0.009	-0.019	0.017
Age	0.002	0.010	-0.017	0.021
Prior therapy	0.006	0.022	-0.039	0.050
Treatment	-0.145	0.190	-0.524	0.234
log.shape1	0.375	2.123	-3.871	4.620
log.shape2	0.440	0.274	-0.108	0.989
log.shape3	-1.075	0.634	-2.344	0.193
Simulation-Sampling Importance Resampling				
	Mean	SD	LB	UB
Intercept	-0.060	1.463	-1.867	2.861
Karno	0.040	0.006	0.031	0.049
Celltype	-0.045	0.078	-0.210	0.092
Diagtime	-0.001	0.007	-0.014	0.017
Age	0.005	0.008	-0.014	0.019
Prior therapy	0.006	0.021	-0.039	0.047
Treatment	-0.046	0.222	-0.433	0.387
shape1	3.776	5.964	0.251	20.792
shape2	1.866	0.324	1.254	2.389
shape3	0.274	0.103	0.086	0.508

Table 4.5: *Posterior summary of lung cancer data by LaplaceApproximation function, giving two summary first from Nelder-mead method and second from sampling importance resampling under the assumption of Weibull Lomax model.*

4.7.1 Fitting with LaplacesDemon function

Now, the data has been fitted using `LaplacesDemon` function for Weibull Lomax distribution using independent Metropolis algorithm. The simulated posterior summary is being summarized in Table 4.6, and its trace plots, posterior density plots and auto-correlation plots are reported in Figure 4.11.

```
Initial.Values<-as.initial.values(M1)
M2<-LaplacesDemon(Model, Data=MyData, Initial.Values,
    Covar=M1$Covar, Iterations=20000, Status=100, Thinning=1,
    Algorithm="IM",Specs=list(mu=M1$Summary1[1:length(Initial.Values),1]))
M2
```

	Mean	SD	MCSE	ESS	LB	Median	UB
Intercept	0.974	0.816	0.049	418.738	-0.698	0.937	2.617
Karno	0.038	0.003	0.000	480.427	0.032	0.038	0.043
Cell type	-0.074	0.051	0.003	527.950	-0.167	-0.071	0.025
Diagtime	-0.001	0.006	0.000	481.723	-0.012	-0.001	0.010
Age	0.003	0.006	0.000	395.196	-0.008	0.003	0.014
Prior therapy	0.006	0.013	0.001	617.846	-0.021	0.006	0.032
Treatment	-0.136	0.106	0.006	498.795	-0.339	-0.142	0.073
shape1	2.442	2.362	0.141	450.176	0.464	1.573	8.836
shape2	1.626	0.215	0.013	484.064	1.244	1.615	2.101
shape3	0.313	0.090	0.006	381.647	0.165	0.304	0.505

Table 4.6: *Simulated posterior summaries of lung cancer data under the assumption of Weibull Lomax distribution.*

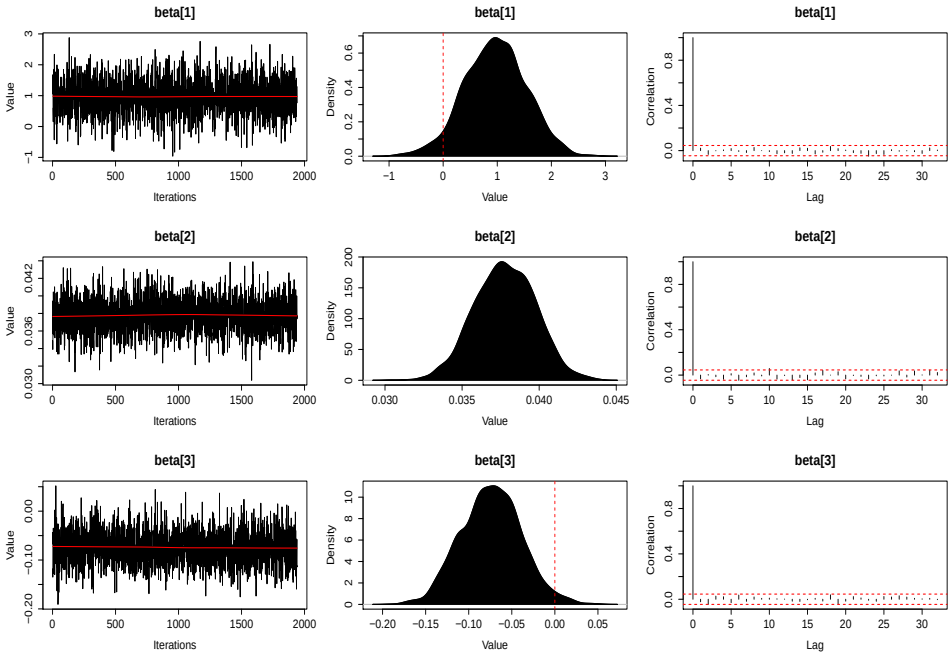

Figure 4.11: *Simulated posterior density plots of the parameter of the distribution. The leftmost is the trace plot, the middlemost is the density plot and the rightmost is the auto correlation plot, showing low autocorrelation at different lags.*

4.8 Model comparison

Model selection is the task of choosing appropriate model from a set of candidate models. Here, Table 4.7 clearly shows that exponential Lomax is the appropriate model for veteran data as it has minimum value of DIC and deviance as compared to Lomax and Weibull Lomax. DIC and deviance are very good criteria of model comparison as they have the potential to provide powerful comparison of complex models.

Models	Deviance	DIC
Lomax	1446.665	1448.316
Exponential Lomax	1403.089	1405.332
Weibull Lomaxl	1440.137	1442.882

Table 4.7: *Model comparison of Lomax, Weibull Lomax and exponential Lomax models for the lung cancer data. It is evident from this table that exponential Lomax fits much better than Weibull Lomax and Lomax.*

4.9 Conclusion

In this chapter, Bayesian approach has been employed to model the real survival data under the assumption of Lomax and its extended forms namely, exponential Lomax and Weibull Lomax distribution. These distributions have been used as a Bayesian model to fit the survival data. This chapter includes the derivation of joint and marginal posterior densities of these three models. Asymptotic approximation and simulation methods, the two most important techniques have been implemented to solve the high-dimensional integrations. These two methods have been implemented using the functions of `LaplacesDemon` package. The function `LapalaceApproximation` is the main function for the purpose of optimization in Bayesian scenario whereas `LaplacesDemon` is the function which is meant for implementation of Markov chain Monte Carlo simulation tools. The central part of the chapter has been composed of the description of `R` code. After Bayesian modeling of these distributions, the last step is to compare the goodness of fit of the models through the values of DIC and deviance, as per recommendation of Gelman et al., (2004) deviance is the best criteria for model selection. Following Table 4.7, it could be noticed that the value of DIC and deviance for exponential Lomax distribution is the least value followed by Weibull Lomax and Lomax. Hence, it could be concluded that the EL is highly competitive in the sence of fitting real survival data.

The code developed in `R` can be used in other areas of regression modeling besides in the field of survival, because of their general nature and paradigm. Finally, Bayesian approach is more suitable even if sample size is small and it can be used very effectively in the modeling of survival data wherein non Guassian model like Weibull, Lomax, exponential Lomax and Weibull Lomax commonly fit. Undoubtedly, `R` enhances the astonishing vigour of Bayesian approach.

Bibliography

Abdul-Moniem, I. B., and Abdel-Hameed, H. F.(2012). On exponentiated Lomax distribution, *International Journal of Mathematical Archive* **3**, 2144-2150.

Ahmad, A. A. (2001). Moments of order statistics from doubly truncated continuous distribution and characterizations. *Statistics* **35**, 479-494.

Aitchison, J., and Dunsmore, I. R. (1975). *Statistical Prediction Analysis*. Cambridge University Press, Cambridge.

Albert, J. (2009). *Bayesian Computation with R*, Second Edition. Springer, New York.

Alshunnar, F. S., Raqab, M., and Kundu, D. (2010). On the comparison of the Fisher information of the log-normal and generalized Rayleigh distributions. *Journal of Applied Statistics* **37**, 391-404.

Aludaat, K. M., Alodat, M. T., and Alodat, T. T. (2008). Parameter estimation of Burr type X distribution for grouped data. *Applied Mathematical Science* **2**, 415-423.

Anderson, P. K., Borgan, O., Gill, R. D., and Keiding, N. (1993). *Statistical Models Based on Counting Processes.* New York: Springer-Verlag.

Anscombe, F., and Aumann, R. (1963). A definition of subjective probability. *The Annals of Mathematical Statistics* **34**, 199-205.

Ashour, S. K., and Afify, W. M. (2007). Statistical analysis of exponentiated Weibull family under type I progressive interval with random removals. *Applied Sciences Research* **3**, 1851-1863.

Ashour, S. K., and Afify, W. M. (2008). Estimation of the parameters of exponentiated Weibull family with type II progressive interval censoring with random removals, *Applied Sciences Research* **4**, 1428-1442.

Azevedo-Filho, A., and Shachter, R. (1994). Laplace's Method Approximations for Probabilistic Inference in Belief Networks with Continuous Variables. *In Uncertainty in Artificial Intelligence* Mantaras, R. and Poole, D., Morgan Kauffman, San Francisco, CA, 28-36.

Bayes, T., and Price, R. (1763). An sssay towards solving a problem in the doctrine of chances. By the late Rev. Mr. Bayes, communicated by Mr. Price, in a letter to John Canton, MA. and F.R.S. *Philosophical Transactions of the Royal Society of London* **53**, 370-418.

Bennett, S. (1983). Log-logistic regression models for survival data. *Applied Statistics* **32**, 165-171.

Berry, D. A. (1996). *Statistics: A Bayesian Perspective*. Duxbery Press, Belmont, CA.

Berger, J. O. (1993). *Statistical Decision Theory and Bayesian Analysis*, Revised Second Edition. Springer-Verlag, New York.

Berger, J. O. (1985). *Statistical Decision Theory and Bayesian Analysis*, 2nd Ed. Springer-Verlag, New York.

Berger, J. O. (2006). The case for objective Bayesian analysis. *Bayesian Analysis* **1**, 385-402.

Bernardo, J. M., and Smith, A. F. M. (1994). *Bayesian Theory*. Wiley, Chichester, England.

Bernardo, J. M., and Smith, A. F. M. (2000). *Bayesian Theory*. John Wiley and Sons, Chichester, West Sussex.

Besag, J., Green, P., Higdon, D., and Mengersen, K. (1995). Bayesain computation and stochastic systems. *Statistical Science* **10**, 3-66.

Box, G. E. P., and Tiao, G. C. (1973). *Bayesian Inference in Statistical Analysis*. John Wiley and Sons, New York.

Broyden, C. G. (1970). The convergence of a class of double rank minimization algorithms: 2. The new algorithm. *Journal of the Institute of Mathematics and its Applications* **6**, 76-90.

Burr, I. W. (1942). Cumulative frequency distribution, *Annals of Mathematical Statistics* **13**, 215-232.

Campbell, G., and Ratnaparkhi, M. V. (1993). An application of Lomax distributions in receiver operating characteristic (ROC) curve analysis. *Communications in Statistics-Theory Methods* **22**, 1681-1697.

Carlin, B. P., and Louis, T. A. (1996). *Bayes and Empirical Bayes Methods for Data Analysis*. Chapman & Hall, London.

Carlin, B. P., and Louis, T. A. (2008). *Bayesian Methods for Data Analysis*, Third Edition. Chapman and Hall/CRC, Boca Raton, FL.

Casella, G. and George, E. (1992). Explaining the Gibbs sampler, *The American Statistician* **46**, 167-174.

Chahkandi, M., and Ganjali, M. (2009). On some lifetime distributions with decrasing failure rate. *Computational Statistics and Data Analysis* **53**, 4433-4440.

Chen, K., Jin, Z., and Ying, Z. (2002). Semiparametric analysis of transformation models with censored data. *Biometrika* **89**, 659-668.

Chen, M. H., Shao, Q. M., and Ibrahim, J. G. (2000). *Monte Carlo Methods in Bayesian Computation*. New York: Springer.

Chib, S., and Greenberg, E. (1995). Understanding the Metropolis-Hastings algorithm, *American Statistician* **49**, 327-335.

Christensen, R., Johnson, W., Branscum, A., and Hanson, T. E. (2011). *Bayesian Ideas and Data Analysis*. Chapman & Hall, London.

Collet, D. (1994). *Modelling Survival Data in Medical Research*. Chapman & Hall, London.

Collet, D. (2003). *Modelling Survival Data in Medical Research*, second edition. Chapman & Hall, London.

Cordeiro, G. M., Ortega, E. M. M., and Popovi'c, B. V. (2013). The gamma-Lomax distribution. *Journal of Statistical computation and Simulation* **85**, 305-319.

Congdon, P. (2001). *Bayesian Statistical Modelling*. John Wiley and Sons, Chichester, West Sussex.

Congdon, P. (2003). *Applied Bayesian Modelling*. Wiley, Chichester, England.

Cox, D. R., and Oakes, D. (1984). *Analysis of Survival Data*. Landon: Chapmann & Hall.

Cox, D. R. (1975). Partial likelihood. *Biometrika* **62**, 269-276.

Damien, P., Wakefield, J., and Walker, S. (1999). Gibbs sampling for Bayesian non-conjugate and hierarchical models by using auxiliary variable, *Journal of the Royal Statistical Society B* **61**, 331-344.

De Finetti, B. (1931). Probabilismo. *Erkenntnis* **31**, 169-223. English translation as Probabilism: A Critical essay on the theory of probability and on the value of science.

De Finetti, B. (1937). La Prevision: ses lois logigues, ses sources subjectives, Annales de l'Institut Henri Poincare. English translation in H.E. Kyburg and H.E. Smokler (eds), (1964). Foresight: its logical laws, its subjective sources. *Studies in Subjective Probability*. New York: Wiley.

De Finetti, B. (1974). *Theory of Probability*. **1**. John Wiley and Sons, New York.

De Finetti, B. (1975). *Theory of Probability*. **2**. John Wiley and Sons, New York.

DeGroot, M. H. (1970). *Optimal Statistical Decisions*. McGraw-Hill, New York.

Efron, B. (1977). The efficiency of Cox's likelihood function for censored data. *Journal of the American Statistical Association* **72**, 557-565.

Elshahat, M. A. T. (2006). Bayesian estimation of the parameters of the exponentiated Weibull distribution with progressively type I interval censored sample. *Journal of Commerce Research* Faculty of Commerce, Zagazig University, Egypt, **28**, 5-21

Elshahat, M. A. T. (2008). Approximate Bayes estimators for the exponentiated Weibull parameters with progressive interval censoring, *The 20th Annual Conference on Statistics & Modeling In Human & Social Science* Faculty of Economic & Political Science, Cairo University, Egypt, 123-136.

Epstein, B. (1954). Truncated life tests in the exponential case. *Annals of Mathematical Statistics* **25**, 555-564.

Epstein, B. (1960). Statistical life test acceptance procedure. *Technometrics* **2**, 435-446.

Epstein, B., and Sobel, M. (1953). Life testing. *Journal of the American Statistical Association* **48**, 486-502.

Epstein, B., and Sobel, M. (1954). Some theorems relevant to life testing from an exponential distribution. *Annals of Mathematical Statistics* **25**, 373-381.

Epstein, B., and Sobel, M. (1955). Sequential life tests in the exponential case. *Annals of Mathematical Statistics* **26**, 82-93.

Erkanli, A. (1994). Laplace approximations for posterior expectation when the model occurs at the boundary of the parameter space, *Journal of the American Statistical Association* **89**, 205-258.

Feller, W. (1968). *An Introduction to Probability Theory and its Applications*, third edition. **1**, New York: Wiley.

Fleming, T. R., and Harrington, D. P. (1991). *Counting Processes and Survival Analysis*. New York: Wiley.

Fletcher, R. (1970). A new approach to variable metric algorithms. *Computer Journal* **13**, 317-322.

Gamerman, D., and Lopes, H. (2006). *Markov Chain Monte Carlo, Texts in Statistical Science*, 2nd ed., Chapman & Hall, New York.

Geisser, S. (1993). *Predictive Inference: An Introduction*. Chapman and Hall, New York.

Gelfand, A., Hills, S., Racine-Poon, A., and Smith, A. (1990). Illustration of Bayesian inference in normal data models using Gibbs sampling. *Journal of the American Statistical Association* **85**, 972-985.

Gelfand, A., and Smith, A. (1990). Sampling-based approaches to calculating marginal densities. *Journal of the American Statistical Association* **85**, 398-409.

Gelman, A., Carlin, J. B., Stern, H. S., and Rubin, D. B. (1995). *Bayesian Data Analysis*. Chapman & Hall, London.

Gelman, A., Carlin, J. B., Stern, H. S., and Rubin, D. B. (2004). *Bayesian Data Analysis*, Second Edition. Chapman and Hall/CRC, Boca Raton, FL.

Gelman, A., Carlin, J. B., Stern, H. S., Dunson, D. B., Vehtari, A., and Rubin, D. B. (2014). *Bayesian Data Analysis*, Third Edition. Chapman and Hall/CRC, Boca Raton, FL.

Gelman, A. and Hill, J. (2007). *Data Analysis Using Regression and Multileve/Hierarchical Models,* Cambridge University Press, New York.

Geman, S., and Geman, D. (1984). Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **6**, 721-741.

George, E. I., and McCulloch, R. E. (1993). Variable selection via Gibbs sampling. *Journal of the American Statistical Association* **88**, 881-889.

Geweke, J. (1989). Bayesian inference in econometric models using Monte Carlo integration. *Econometrica* **57**, 1317-1339.

Ghitany, M. E., AL-Awadhi, F. A., and Alkhalfan, L. A. (2007). Marshall-Olkin extended Lomax distribution and its applications to censored data. *Communications in Statistics-Theory and Methods* **36**, 1855-1866.

Gilks, W., Richardson, S., and Spiegelhalter, D. (1996). *Markov Chain Monte Carlo in Practice, Interdisciplinary Statistics*. Chapman & Hall, Suffolk, UK.

Givens, G., and Hoeting, J. (2005). *Computational Statistics, Wiley Series in Probability and Statistics*. Wiley, Hoboken, NJ.

Glanzel, W. (2008). On some new bibliometric applications of statistics related to the h-index. *Scientometrics* **77**, 187-196.

Goldfarb, D. (1970). A family of variable metric methods derived by variational means. *Mathematics of Computation* **24**, 23-26.

Goldstein, M. (2006). Subjective Bayesian analysis: principles and practice. *Bayesian Analysis* **1**, 403-420.

Gordon, N. J., Salmond, D. J., and Smith, A. F. M. (1993). Novel approach to nonlinear/non-Gaussian Bayesian state estimation. *IEE Proceedings-F* **140**, 107-113.

Green, P. (1995). Reversible jump Markov chain Monte Carlo computation and Bayesian model determination, *Biometrika* **82**, 711-732.

Greenhouse, J., and Wasserman, L. (1996). A practical, robust method for Bayesian model selection: a case study in the analysis of clinical trials. *Institute of Mathematical Statistics Lecture Notes-Monograph Series 29*, 41-62.

Grieve, A. P. (1987). Applications of Bayesian software: two examples. *The Statistician* **36**, 283-288.

Gupta, R. C., Ghitany, M. E., and Al-Mutairi, D. K. (2010). Estimation of reliability from Marshall-Olkin extended Lomax distributions. *Journal of Statistical Computation and Simulation* **80**, 937-947.

Gupta, R. D., and Kundu, D. (1999). Generalized exponential distributions, *Australian and New Zealand Journal of Statistics* **41**, 173-188.

Gupta, R. D., and Kundu, D. (2001). Exponentiated exponential family: an alternative to gamma and Weibull distributions. *Biometrical Journal* **43**, 117-130.

Gupta, R. D., and Kundu, D. (2007). Generalized exponential distribution: existing results and some recent development. *Journal of Statistical Planning and Inference* **137**, 3537-3547.

Gupta, R. C., Gupta, P. L., and Gupta, R. D. (1998). Modeling failure time data by Lehmann alternatives, *Communications in Statistics-Theory and Methods* **27**, 887-904.

Gurvich, M. R., Dibenedetto, A. T., and Rande, S. V. (1997). A new statistical distribution for characterizing the random strength of brittle materials. *Journal of Materials Science* **32**, 2559-2564.

Harris, C. M. (1968). The Pareto distribution as a queue service descipline. *Operations Research* **16**, 307-313.

Hassan, A. S., and Al-Ghamdi, A. S. (2009). Optimum step stress accelerated life testing for Lomax distibution. *Journal of Applied Sciences Research* **5**, 2153-2164.

Hastings, W. (l970). Monte Carlo sampling methods using Markov chains and their applications, *Biometrika* **57**, 97-109.

Higdon, D. (1998). Auxiliary variable methods for Markov chain Monte Carlo with applications. *Journal of the American Statistical Association* **93**, 585-595.

Holland, O., Golaup, A., and Aghvami, A. H. (2006). Traffic characteristics of aggregated module downloads for mobile terminal reconfiguration, *IEE proceedings on Communications* **135**, 683-690.

Ibrahim, J. G., Chen, M.H., and Sinha, D. (2001). *Bayesian Survival Analysis*. New York: Springer-Verlag.

Ibrahim, J. G., and Laud, P. W. (1994). A predictive approach to the analysis of designed experiments. *Journal of the American Statistical Association* **89**, 309-319.

Ibrahim, J. G., and Chen, M. H. (1998). Prior distributions and Bayesian computation for proportional hazards models. *Sankhya* **60**, 48-64.

Ibrahim, J. G., Chen, M. H., and MacEachern, S. N. (1999). Bayesian variable selection for proportional hazards model. *Canadian Journal of Statistics* **27**, 701-717.

Jaheen, Z. F., and Al-Matrafi, B. N. (2002). Bayesian prediction bounds from the scaled Burr type X model. *Computation and Mathematical Application* **44**, 587-594.

Jeffreys, H. (1961). *Theory of Probability*, Third Edition. Oxford University Press, London.

Jiang, R., and Murthy, D. N. P. (1999). The exponentiated Weibull family: a graphical approach, *IEEE Transactions on Reliability* **48**, 68-72.

Kalbfleisch, J. D., and Prentice, R. L. (1980). *The Statistical Analysis of Failure Time Data*. New York: John Wiley and Sons.

Kalbfleisch, J. D., and Prentice, R. L. (2002). *The Statistical Analysis of Failure Time Data*, second edition, New York: John Wiley and Sons.

Kalbfleisch, J. D., and MacKay, R. J. (1978). *Censoring and the Immutable likelihood*. University of Waterloo. Department of Statistics STAT-78-09, Technical Report.

Kass, R. E., Tierney, L., and Kadane, J. B. (1989). Approximate methods for assessing influence and sensitivity in Bayesian analysis. *Biometrika* **76**, 663-674.

Kass, R., Carlin, B., Gelman, A., and Neal, R. (l998). Markov chain Monte Carlo in practice: A roundtable discussion, *The American Statistician* **52**, 93-100.

Kass, R. E., and Raftery, A. E. (1995). Bayes factor. *Journal of the American Statistical Association* **90**, 773-795.

Khan, Y., Akhtar, M. T., Shehla, R., and Khan, A. A. (2015). Bayesian modelling of forestry data with R using optimization and simulation tools. *Journal of Applied Analysis and Computation* **5**, 38-51.

Khan, Y., and Khan, A. A. (2014).Bayesian regression analysis of Weibull model with R and Bugs. *Statistical Methodologies and Applications* **2**, 131-140

Khan, Y., and Khan, A. A. (2013a). Bayesian analysis of Weibull and log-normal survival models with censoring mechanism. *International Journal of Applied Mathematics* **26**, 671-683.

Khan, Y., and Khan, A. A. (2013b) .Bayesian survival analysis of regression model using Weibull. *International Journal of Innovative Research and Science, Engineering and Technology* **2**, 7199-7204.

Kim, C., Jung, J., and Chung, Y. (2009). Bayesian estimation for the exponentiated Weibull model under type II progressive censoring. *Statistical Papers* **51**, 375-387.

Klein, J. P., and Moeschberger, M. L. (2003). *Survival Analysis: Techniques for Censored and Truncated Data*. New-York: Springer-Verlag.

Krall, J. M., Uthoff, V. A., and Harley, J. B. (1975). A set-up procedure for selecting variables associated with survival. *Biometrics* **31**, 49-57.

Kumar, S. K., Rajkumar, S. V., Dispenzieri, A., Lacy, M. Q., Hayman, S. R., Buadi, F. K., Zeldenrust, S. R., Dingli, D., Russell, S. J., Lust, J. A., Greipp, P. R., Kyle,R. A., and Gertz, M. A. (2008). Improved survival in multiple myeloma and the impact of novel therapies. *The American Society of Hematology* **111**, 2516-2520

Kundu, D., and Gupta, R. D. (2004). Characterizations of the proportional (reversed) hazard class. *Communications in Statistics-Theory and Methods* **33**, 3095-3102.

Kundu, D., and Raqab, M. Z. (2005). Generalized Rayleigh distribution: different methods of estimations. *Computational Statistics and Data Analysis* **49**,187-200.

Kundu, D., and Raqab, M. Z. (2007). Discriminating between the generalized Rayleigh and log-normal distribution. *Statistics* **41**, 505-515.

Kuo, L., and Yang, T. Y. (1996). Bayesian computation for non-homogeneous Poisson processes in software reliability. *Journal of the American Statistical Association* **91**, 763-773.

Lalitha, S., and Mishra, A. (1996). Modified maximum likelihood estimation for Rayleigh distribution. *Communications in Statistics-Theory and Methods* **25**, 389-401.

Laplace, P. (1774). Memoire sur la Probabilite des Causes par les Evenements. *l'Academie Royale des Sciences* **6**, 621-656. English translation by S.M. Stigler in 1986 as Memoir on the Probability of the causes of events in *Statistical Science* **1**, 364-378.

Laplace, P. (1814). Essai Philosophique sur les Probabilites. English translation in Truscott, F.W. and Emory, F. L. (2007) from (1902) as A philosophical essay on probabilities (1840). ISBN 1602063281, translated from the French 6th ed.

Lawless, J. F. (1982). *Statistical Models and Methods for Life Time Data*. New York: Wiley.

Lawless, J. F. (2003). *Statistical Models and Methods for Life Time Data*. New York: Wiley.

Leathem, A. J., and Brookes, S. A. (1987). Predictive values of lectin binding on breast cancer recurrence and survival. *The Lancet* **I**, 1054-1056.

Lee, E. T. (1992). *Statistical Methods for Survival Data Analysis*. New York: Wiley.

Lemonte, A. J., and Cordeiro, G. M. (2013). An extended Lomax distribution. *Statistics: A Journal of Theoretical and Applied Statistics* **47**, 800-816.

Lindley, D. V. (1971). *Bayesian Statistics: A Review*. SIAM, Philadelphia.

Lindley, D. V. (1980). Approximate Bayesian method. *Trabjos de Estadistic*. 3**1**, 223-237.

Lomax, K. S. (1954). Business failures: Another example of the analysis of failure data. *Journal of the American Statistical Association* **49**, 847-852.

Madigan, D., and Raftery, A. E. (1994). Model selection and accounting for model uncertainty in graphical models using Occam's window. *Journal of the American Statistical Association* **89**, 1535-1546.

Malinowska, I., and Szynal, D .(2005). Inference and prediction for the Burr type X model based on the kth lower record. *Journal of Statistical Theory and Applications* **4**, 282-291.

Marin, J. M., and Robert, C. P. (2007). *Bayesian Core: A Practical Approach to Computational Bayesian Statistics*. Springer-Verlag, New York.

Marshall, A. N., and Olkin, I. (1997). A new method for adding a parameter to a family of distributions with applications to the exponential and Weibull families. *Biometrika* **84**, 641-652.

Metropolis, N., Rosenbluth, A, Rosenbluth, M., Teller, A., and Teller, E. (1953). Equations of state calculations by fast computing machine. *Journal of chemical Physics* **21**, 1087-1092.

Metropolis, N., and Ulam, S. (l949). The Monte Carlo method, *Journal of the American Statistical Association* **44**, 335-341.

Merller, J. (1999). Perfect simulation of conditionally specified models. *Journal of the Royal Statistical Society B* **61**,251-264.

Montazer, H. A., and Shayib, M. A. (2010). Shrinkage estimators for calculating reliability, Weibull case. *Journal of Applied Statistical Science* **17**, 219-233.

Mousa, M. A. (2001). Inference and Prediction for the Burr type X model based on records. *Statistics* **35**, 415-425.

Mudholkar, G. S., and Srivastava, D. K. (1993). Exponentiated Weibull family for analyzing bathtub failure-rate data. *IEEE Transactions on Reliability* **42**, 299-302.

Mudholkar, G. S., Srivastava, D. K., and Fraimer, M. (1995). The exponentiated Weibull family: A reanalysis of the bus-motor-failure data. *Technometrics* **37**, 436-445.

Mudholkar, G. S., and Hutson, A. D. (1996). The exponentiated Weibull family: Some properties and a flood data application. *Communications in Statistics-Theory and Methods* **25**, 3059-3083.

Muller, P., and Rios Insua, D. (1998). Issues in Bayesian analysis of neural network models. *Neural Computation* **10**, 571-592.

Muller, P., and Roeder, K. (1997). A Bayesian semiparametric model for case-control studies with errors in variables. *Biometrika* **84**, 523-537.

Murphy, S. A., Rossini, A. J., and Vaart, A. W. V. D. (1997). Maximum likelihood estimation in the proportional odds model. *Journal of the American Statistical Association* **92**, 968-976.

Murthy, D. N. P., Xi, M., and Jiangs, R. (2004). *Weibull model*. Wiley, Hoboken.

Nadarajah, S. (2009). Bathtub-shaped failure rate functions. *Quality and Quantity* **43**, 855-863.

Nassar, M. M., and Eissa, F. H. (2003). On the exponentiated Weibull distribution. *Communications in Statistics-Theory and Methods* **32**, 1317-1336.

Nassar, M. M., and Eissa, F. H. (2004). Bayesian estimation for the exponentiated Weibull model. *Communications in Statistics-Theory and Methods* **33**, 2343-2362.

Neal, R. (2003). Slice sampling, *The Annals of Statistics* **31**, 705-767.

Neal, R. (1998). Suppressing random walks in Markov chain Monte Carlo using ordered overrelaxation, in *M. Jordan, ed., Learning in Graphical Models, Kluwer Academic Publishers*, Dordrecht, 205-230.

Neal, P., and Roberts, G. (2008). Optimal scaling for random walk Metropolis on spherically constrained target densities. *Methodology and Computing in Applied Probability* **10**, 277-297.

Nelder, J. A., and Mead, R. (1965). A simplex method for function minimization. *The Computer Journal* **7**, 308-313.

Newton, M. A., Czado, C., and Chappell, R. (1996). Bayesian inference for semi-parametric binary regression. *Journal of the American Statistical Association* **91**, 142-153.

Nikulin, M., and Haghighi, F. (2006). A Chi-squared test for the Generalized Power Weibull family for the head-and-neck cancer censored data. *Journal of Mathematical Sciences* **133**, 1333-1341.

Nocedal, J., and Wright, S. J. (1999). *Numerical Optimization*. Springer-Verlag.

Ntzoufras, I. (2009). *Bayesian modelling using WinBugs*. John Wiley & Sons.

O'Hagan, A. (1994). *Bayesian Inference. Kendall's Advanced Theory of Statistics* **2B**. Halsted Press, New York.

Peto, R., Pike, M. C ., Armitage, P., BresloW, N. E., Cox, D, R., Howard, S. V., Mantel, N., McPherson, K., Peto, J., and Smith P. G. (1976). Design and analysis of randomized clinical trials requiring prolonged observation of each patient. Part I: Introduction and design. *British Journal of Cancer* **34**, 585-612.

Peto, R., Pike, M. C., Armitage, P., Breslow, N. E., Cox, D, R., Howard, S. V., Mantel, N., McPherson, K., Peto, J., and Smith P. G. (1977). Design and analysis of randomized clinical trials requiring prolonged observation of each patient. Part II: Analysis and examples. *British Journal of Cancer* **35**, 1-39.

Plummer, M. (2003). JAGS: A program for analysis of Bayesian graphical models using Gibbs sampling. Currently at http://mcmc-jags.sourceforge.net

Prentice, R. L. (1973). Exponential survivals with censoring and explanatory variables. *Biometrika* **60**, 279-284.

Propp, J., and Wilson, D. (1996). Exact sampling with coupled Markov chains and applications to statistical mechanics. *Random Structures and Algorithms* **9**, 223-252.

R Core Team (2015) R: A language and environment for statistical computing. *R Foundation for Statistical Computing* Vienna, Austria. URL http://www.Rproject.org/

Raiffa, H., and Schlaifer, R. (1961). *Applied Statistical Decision Theory*. Division of Research, Graduate School of Business Administration, Harvard University, Boston.

Ramsey, F. (1926). Truth and Probability. *The Foundations of Mathematics and other Logical Essays* Harcourt, Brace and Company, New York.

Raqab, M. Z. (1998). Order statistics from the Burr type X model. *Computers and Mathematics with Applications* **36**, 111-120.

Raqab, M. Z., and Kundu, D. (2006). Burr type X distribution: revisited. *Journal of Probability and Statistical Science* **4**, 179-193.

Rayleigh, J. W. S. (1880). On the resultant of a large number of vibration of the somepith and of arbitrary phase. *Philosophical Magazine*, 5th series, **10**, 73-78.

Riedmiller, M. (1994). Advanced supervised learning in multi-layer perceptrons from backpropagation to adaptive learning algorithms. *Computer Standards and Interfaces* **16**, 265-278.

Rinne, H. (2009). *The Weibull distribution.* a handbook of CRC Press, Boca Raton.

Roberts, G., Gelman, A., and Gilks, W. (1997). Weak convergence and optimal scaling of random walk Metropolis algorithms. *The Annals of Statistics* **7**, 10-120.

Robert, C. P. (2007). *The Bayesian Choice: From Decision-Theoretic Foundations to Computational Implementation*, Second Edition. Springer: New York.

Robert, C., and Casella, G. (2004). *Monte Carlo Statistical Methods*, Springer Texts in Statistics, 2nd ed., Springer-Verlag, New York.

Robert, C. P. (2001). *The Bayesian Choice*, 2nd Ed. Springer-Verlag, New York.

Roberts, G. and Rosenthal, J. (2001). Optimal scaling for various Metropolis-Hastings algorithms. *Statistical Science* **16**, 351-367.

Rosenthal, J. S. (1995). Minorization conditions and convergence rates for Markov chain Monte Carlo. *Journal of the American Statistical Association* **90**, 558-566.

Rossi. P., Allenby. G., and McCulloch. R. (2005). *Bayesian Statistics and Marketing.* John Wiley & Sons, West Sussex, England.

Rubinstein, R. Y., and Kroese, D. P (2008). *Simulation and the Monte Carlo Methods.* Jhon Wiley and Sons.

Sargent, D. J. (1997). A flexible approach to time varying coefficients in the Cox regression setting. *Lifetime Data Analysis* **3**, 13-25.

Sargent, D. J. (1998). A general framework for random effects survival analysis in the Cox proportional hazards setting. *Biometrics* **54**, 1486-1497.

Sartawi, H. A., and Abu-Salih, M. S. (1991). Bayes prediction bounds for the Burr type X model. *Communications in Statistics-Theory and Methods* **20**, 2307-2330.

Savage, L. J. (1954). *The Foundations of Statistics*. John Wiley and Sons, New York.

Savage, L. J. (1972). *The Foundations of Statistics*, 2nd revised Ed. Dover, New York.

Shanno, D. F. (1970). Conditioning of quasi-Newton methods for function minimization. *Mathematics of Computation* **24**, 647-650.

Siddiqui, M. M., (1962). Some problems conectede with Rayleigh distributions. *Journal of Research of the National Bureau of Standards* **60D**, 167-174.

Sinha, S. K., and Howlader, H. A. (1983). Credible and HPD intervals of the parameter and reliability of Rayleigh distribution. *IEEE Transactions on reliability* **32**, 217-220.

Singh, U., Gupta, P. K., and Upadhyay, S. K. (2002). Estimation of exponentiated Weibull shape parameters under Linex loss function. *Communications in Statistics-Simulation and Computation* **31**, 523-537.

Singh, U., Gupta, P. K., and Upadhyay, S. K. (2005a). Estimation of parameters for exponentiated-Weibull family under type II censoring scheme. *Computational Statistics and Data Analysis* **48**, 509-523.

Singh, U., Gupta, P. K., and Upadhyay, S. K. (2005b). Estimation of three-parameter exponentiated-Weibull distribution under type II censoring. *Journal of Statistical Planning and Inference* **134**, 350-372.

Singh, U., Gupta, P. K., and Upadhyay, S. K. (2006). Some point estimates for the shape parameters of exponentiated Weibull famliy. *Journal of the Korean Statistical Society* **35**, 63-77.

Smith, A., and Roberts, G. (1993). Bayesian computation via the Gibbs sampler and related Markov chain Monte Carlo methods. *Journal of the Royal Statistical Society B* **55**, 3-23.

Spiegelhalter, D., Thomas, A., Best, N., and Gilks, W. (1996). BUGS 0.5: *Bayesian Znference Using Gibbs Sampling Manual*, MRC Biostatistics Unit, Institute of Public Health, Cambridge, UK.

Spiegelhalter, D., Best, N., Carlin, B., and van der Linde, A. (2002). Bayesian measures of model complexity and fit (with discussion). *Journal of the Royal Statistical Society B* 64, 583-639.

Spiegelhalter, D., Thomas, A., Best, N., and Lunn, D. (2003). WinBUGS User Manual, Version 1.4, MRC Biostatistics Unit, Institute of Public Health and Department of Epidemiology and Public Health, Imperial College School of Medicine, UK, available at http : //www .mrc-bsu. cam.ac . uk/bugs.

Statisticat, LLC. (2015). LaplacesDemon: complete environment for Bayesian inference. R package version 13.03.04, http://www.bayesian-inference. com/software.

Sukhatme, P.V. (1937). Tests of significance for samples of the chi-square population with two degrees of freedom. *Annals of Eugenics* **8**, 52-56.

Surles, J. G., and D'Ambrosio, D. M. (2004). A Burr type X chain-of-links model for carbon fibers. *Journal of Composite Materials* **38**, 1337-1343

Surles, J. G., and Padgett, W.J. (2005). Some properties of a scaled Burr type X distribution. *Journal of Statistical Planning and Inference* **128**, 271-280

Tableman, M., and Kim, J. S. (2004). *Survival Analysis using S: Analysis of Time-to-Event Data*. Chapman & Hall/CRC, Boca Raton.

Tahir, M. H., Cordeiro, G. M., Mansoor, M., and Zubair, M. (2015). The Weibull-Lomax distribution: properties and applications. *Hacettepe Journal of Mathematics and Statistics* **44**, 461-480.

Tanner, M. A. (1993). *Tools for Statistical Inference: Methods for the Exploration of Posterior Distributions and Likelihood Functions*, third edition. New York: Springer.

Tanner, M., and Wong, W. (1987). The calculation ofthe posterior distributions by data augmentation. *Journal of the American Statistical Association* **82**, 528-549.

Tierney, L. (1994). Markov chains for exploring posterior distributions. *The Annals of Statistics* **22**, 1701-1762.

Tierney, L., and Kadane, J. B. (1986). Accurate approximations for posterior moments and marginal densities. *Journal of the American Statistical Association* **81**, 82-86.

Tierney, L., Kass, R., and Kadane, J., (1989). Fully exponential Laplace approximations to expectations and variances of nonpositive functions. *Journal of the American Statistical Association* **84**,710-716.

Tierney, L. (1998). A note on the Metropolis Hastings algorithm for general state spaces. *Annals of Applied Probability* **8**, 1-9.

Verdinelli, and Wasserman (1998). Bayesian goodness of fit testing using infinite dimensional exponential families. *Annals of Statistics* **26**, 1215-1241.

Waller, L. A., Carlin, B. P., Xia, H., and Gelfand, A. (1997). Hierarchical spatio-temporal mapping of disease rates. *Journal of the American Statistical Association* **92**, 607-617.

Zellner, A. (1971). *An Introduction to Bayesian Inference in Econometrics*. John Wiley and Sons, New York.

Wong, W. H., and Li, B. (1992). Laplace expansion for posterior densities of nonlinear functions of parameters. *Biometrika* **79**, 393-398.

Zhou, M., Yang, D., Wang, Y., and Nadarajah, S. (2008). Moments of the scaled Burr type X distribution. *Journal of Computational Analysis and Applications* **10**, 523-525.